

Perspective-n-Point 問題とその派生問題に対する
安定かつ高速な解法に関する研究

2021年 3月

中野 学

Perspective-n-Point 問題とその派生問題に対する
安定かつ高速な解法に関する研究

中野 学

システム情報工学研究科

筑波大学

2021年 3月

概要

本論文では, Perspective-n-Point (PnP) 問題とその派生問題である PnPf 問題と PnPfr 問題への解法に関する研究成果について報告する. PnP 問題とは, 世界座標系における n 点 (point) の 3 次元点とそれらが画像平面へ射影 (Perspective projection) された画像座標との対応から, 焦点距離などの内部パラメータが校正済みであるピンホールカメラの外部パラメータ, つまり回転行列と並進ベクトルを推定する問題のことである. 幾何学的には最少で $n = 3$ 個の対応点が与えられれば解けることが知られており, P3P 問題と呼ばれている. コンピュータビジョンにおけるもっとも古典的かつ基本的な問題の 1 つであり, 1841 年に最初の解法が発表されて以来, 2000 年代に入ってもなお活発に議論がされている. 外部パラメータだけでなく, 焦点距離やレンズ歪みといった内部パラメータをも推定する PnPf 問題 (f は焦点距離 focal length) や PnPfr 問題 (r は放射方向レンズ歪み radial lens distortion) などの派生問題へと議論は拡張されている. PnP 問題は, リアルタイムで画像から 3 次元解析を行う Visual Simultaneous Localization And Mapping (VSLAM) や Augmented Reality (AR) にて, 画像からカメラの 3 次元位置を推定するモジュールとして利用される. これらのアプリケーション上では, 3 次元点の数や分布の仕方 (平面か非平面) が入力の度が変わるという特性がある. そのため, PnP 問題の解法はリアルタイム処理のみならず様々な課題を同時に解決することが要求される.

本論文は, それらの課題を解決する PnP 問題とその派生問題への解法についてまとめたものであり, 全 8 章で構成されている.

第 1 章では, PnP 問題の研究背景と目的について述べ, 本論文の学術的貢献について説明する.

第 2 章では, まず, PnP 問題がリアルタイム性の高い AR や VSLAM などのアプリケーションにおいて利用されることを述べる. その中で点の数 n が固定である最小解法と n が可変の拡張解法の役割の違いについて説明する. 役割の違いとアプリケーションの性質を踏まえて, 実用の上では安定性, 高速性, 一般性, 拡張性の 4 点が課題であることを明らかにする. そして, それらの課題に対する従来研究の歴史的変遷を説明し, 古典的ながら現在でも精力的に研究が行われていることを説明する.

第 3 章では, PnP 問題に関連が深い研究について紹介する. カメラパラメータ推定問題の観点では, PnP 問題は 1 枚画像を用いたカメラキャリブレーションと解釈できることを指摘する. また, PnP 問題の前提となる対応点の取得方法について説明し, PnP

問題のアプリケーションの具体的な研究例を述べた。最後に、PnP 問題だけでなくコンピュータビジョンにおける幾何学的問題に共通する誤差解析の従来研究を紹介する。

第 4 章では、PnP 問題の理解に必要なカメラ幾何と周辺の数学的な内容について解説する。中でも回転表現の取り扱い、従来手法や本論文における提案手法の導出の違いにも深く関係しているため、各手法を特徴づける重要な要素である。また、平面と非平面の判定方法は Direct Linear Transform (DLT) 法において重要な前処理の一つである。本論文ではもっぱら代数的誤差を扱うが、一般に再投影誤差の最小化が最適と言われているため、それぞれの特性を理解して適宜使い分けることが必要となることを指摘する。

第 5 章では、PnP 問題の最も基礎となる P3P 問題について、特異値分解を要しない新たな直接法を提案する。まず、P3P 問題に対して非直接解法と直接解法の 2 種類が従来提案されてきたことを紹介し、代表的な手法について導出の特徴を紹介する。次に、直接解法の 1 つとして回転行列を射影的奥行き線形結合とみなす新たな手法を提案する。幾何的に複雑な解析を要する従来手法とは異なり、提案手法は代数的な制約条件に基づくため、その導出は非常にシンプルである。シミュレーション実験の結果により、提案手法が 4 次式を用いる手法の中では数値的安定性と処理速度のバランスが最も取れた手法であること実証する。また、従来手法にはアルゴリズムが退化して解が求まらない 3 次元点群とカメラ配置があることを実験的に示し、P3P 問題にはまだ議論の余地があることを示す。

第 6 章では、グレブナー基底を用いて PnP 問題の大域的な最適解を算出する手法を提案する。まず代数的誤差に基づく PnP 問題を非線形最適化問題として定式化し、代表的な従来手法の導出と特徴を簡単に紹介する。次に、未定乗数を用いない最適性条件を導出し、異なる回転表現であってもグレブナー基底が利用できることを示す。シミュレーションデータを用いた比較評価実験により、提案手法は従来不安定とされた Cayley 変換を用いているにも関わらず、四元数を用いた従来手法と同程度に安定であり、かつ、4 倍以上高速であることを実証した。

第 7 章では、PnP, PnPf, PnPfr 問題の 3 問題を同一のアプローチで解く統一的な手法を 2 つ提案する。まず代数的誤差に基づく PnPfr 問題の定式化を行い、PnP 問題と PnPf 問題が包含されることを示す。次に、PnPf 問題と PnPfr 問題に対する従来手法を簡単に説明する。そして、PnPfr 問題を 2 つの部分問題に分割することで、3 つの問題を同一アプローチで解く手法を 2 つ提案する。1 つ目の手法はグレブナー基底を用いて、2 つ目の手法は DLT 法に基づいて、第 1 の部分問題を解く。シミュレーション実験の結果により、提案手法のアプローチは問題なく動作し、3 問題が解けることを実証する。また、PnPfr 問題のシミュレーション実験では提案手法の精度は再投影誤差最小化に劣るものの、実環境ではほぼ同性能を高速に達成するため問題にはならないことを示す。

最後に、第 8 章にて本論文を総括し、残された課題についても述べる。

目次

第 1 章	序論	1
1.1	研究の背景と目的	1
1.2	本論文の学術的貢献	1
1.3	本論文の構成	3
第 2 章	Perspective-n-Point 問題とその派生問題	5
2.1	PnP 問題の役割	5
2.2	PnP 問題の課題	6
2.3	PnP 問題の歴史	7
2.4	第 2 章のまとめ	8
第 3 章	関連研究	9
3.1	カメラキャリブレーション	9
3.2	対応点の検出とマッチング	9
3.3	アプリケーション	10
3.4	ノイズに対する誤差解析	11
3.5	第 3 章のまとめ	12
第 4 章	カメラ幾何	13
4.1	ピンホールカメラモデル	13
4.2	レンズ歪み	14
4.3	内部パラメータの取り扱い	15
4.4	回転行列の表現方法	16
4.4.1	回転行列	16
4.4.2	ロール・ピッチ・ヨー (オイラー角)	17
4.4.3	四元数	17
4.4.4	Cayley 変換	18
4.5	回転行列の補正	18
4.6	平面/非平面の判定	18
4.7	誤差関数	19

4.8	第4章のまとめ	20
第5章	P3P問題への提案手法	21
5.1	P3P問題の定式化	21
5.2	従来手法	21
5.2.1	非直接解法	21
5.2.2	直接解法	23
5.3	提案手法	25
5.3.1	中間座標系への変換	25
5.3.2	射影的奥行き算出	26
5.3.3	世界座標系における外部パラメータの復元	28
5.3.4	Bannoの手法との違い	29
5.4	シミュレーション実験	29
5.4.1	数値的安定性	30
5.4.2	ノイズ頑健性	32
5.4.3	処理速度	32
5.4.4	考察	33
5.5	第5章のまとめ	33
5.6	付録：提案手法の導出詳細	34
第6章	PnP問題への提案手法	37
6.1	PnP問題の定式化	37
6.2	従来手法	38
6.2.1	Direct Linear Transform (DLT) 法	38
6.2.2	交互最適化法	40
6.2.3	EPnP法	40
6.2.4	RPnP法	40
6.2.5	凸最適化に基づく手法	41
6.2.6	グレブナー基底を用いた手法	41
6.3	提案手法	43
6.3.1	未定乗数を用いない1次最適性条件	43
6.3.2	回転表現ごとの最適性条件	44
6.4	シミュレーション実験	46
6.4.1	点数の変化に対するノイズ頑健性	49
6.4.2	画像ノイズの変化に対する頑健性	49
6.4.3	処理速度	50
6.5	考察	50
6.6	第6章のまとめ	50

6.7	付録：グレブナー基底自動計算ツール	52
第7章	PnP、PnPf、PnPfr 問題への統一的な提案手法	59
7.1	PnPfr 問題の定式化	59
7.2	従来手法	60
7.2.1	P4Pf 問題への最小解法	61
7.2.2	PnPf 問題への拡張解法	62
7.2.3	P4Pfr, P5Pfr 問題への最小解法	63
7.3	提案手法 1: VPnPfr 法	64
7.3.1	概要	64
7.3.2	グレブナー基底による部分問題 1 の解法	65
7.3.3	部分問題 2 の解法	67
7.3.4	Root polishing	69
7.4	統一解法 2: EDPnPfr 法	71
7.4.1	非平面	71
7.4.2	平面	73
7.5	実験	73
7.5.1	シミュレーション実験	74
7.5.2	実画像実験	77
7.6	考察	78
7.7	第7章のまとめ	79
7.8	付録：グレブナー基底自動計算ツール	80
第8章	結論	83
	謝辞	85
	参考文献	87
	本研究に関する公表済み論文	95

目次

1.1	Perspective-n-Point (PnP) 問題	2
2.1	PnP 問題の利用シーン	6
3.1	カメラキャリブレーションの例	10
3.2	対応点マッチングの例	10
3.3	PnP 問題が登場するアプリケーション例	11
3.4	焦点距離を高精度に推定するための最良格子パターン	11
4.1	ピンホールカメラモデルにおける射影	14
4.2	Division model に基づくレンズ歪み	15
4.3	再投影誤差と代数的誤差	18
5.1	P3P 問題：Kneip らによる手法の図解	23
5.2	P3P 問題：提案手法における中間座標系	25
5.3	P3P 問題：一般的な点配置における数値演算誤差のヒストグラム	30
5.4	P3P 問題：退化条件における数値演算誤差のヒストグラム	30
5.5	P3P 問題：画像ノイズの変化に対する平均推定誤差	31
6.1	PnP 問題：点の数の変化に対する頑健性評価	47
6.2	PnP 問題：画像ノイズの変化に対する頑健性評価	48
6.3	PnP 問題：点の数の変化に対する平均実行時間	49
7.1	PnPfr 問題：点の数の変化に対する頑健性評価	75
7.2	PnPfr 問題：画像ノイズの変化に対する頑健性評価	76
7.3	PnPfr 問題：点の数の変化に対する平均実行時間	77
7.4	PnPfr 問題：水平画角を変化させた画像に対する歪み除去の結果	78

表目次

1.1	PnP 問題と派生問題における未知パラータと推定に必要な対応点の数 . . .	2
4.1	回転表現の比較	16
5.1	P3P 問題：平均実行時間	32
5.2	P3P 問題：提案手法の実行時間プロファイル	32
6.1	PnP 問題：回転表現の違いによるソルバーの比較	46
7.1	PnPfr 問題：PnPfr 問題への解法の比較	60
7.2	PnPfr 問題：実画像に対する実験結果	77

第 1 章

序論

1.1 研究の背景と目的

本論文では、Perspective-n-Point (PnP) 問題とその派生問題である PnPf 問題と PnPfr 問題への解法に関する研究成果について報告する。PnP 問題とは、図 1.1 に示すように、世界座標系における n 点 (point) の 3 次元点とそれらが画像平面へ射影 (Perspective projection) された画像座標との対応から、焦点距離などの内部パラメータが校正済みであるピンホールカメラの外部パラメータ、つまり回転行列と並進ベクトルを推定する問題のことである。幾何学的には最少で $n = 3$ 個の対応点が与えられれば解けることが知られており、P3P 問題と呼ばれている。コンピュータビジョンにおけるもっとも古典的かつ基本的な問題の 1 つであり、1841 年に最初の解法 [22] が発表されて以来、2000 年代に入ってもなお活発に議論がされている。外部パラメータだけでなく、焦点距離やレンズ歪みといった内部パラメータをも推定する PnPf 問題 (f は焦点距離 focal length) や PnPfr 問題 (r は放射方向レンズ歪み radial lens distortion) などの派生問題へと議論は拡張されている。PnP 問題とその派生問題の関係性を表 1.1 に示す。PnP 問題は、リアルタイムで画像から 3 次元解析を行う Visual Simultaneous Localization And Mapping (VSLAM) や Augmented Reality (AR) にて、画像からカメラの 3 次元位置を推定するモジュールとして利用される。これらのアプリケーション上では、3 次元点の数や分布の仕方 (平面か非平面) が入力の度が変わるという特性がある。そのため、PnP 問題の解法はリアルタイム処理のみならず様々な課題を同時に解決することが要求される。本論文の目的は、それらの課題を解決する PnP 問題とその派生問題への解法を提案することにある。

1.2 本論文の学術的貢献

本論文では PnP 問題とその派生問題について、以下 4 つの手法を提案する。

P3P 問題への解法

特異値分解を用いずに直接に外部パラメータを計算する高速かつ安定な手法。代数的な変数変換に基づくため従来手法よりも導出が簡潔である。

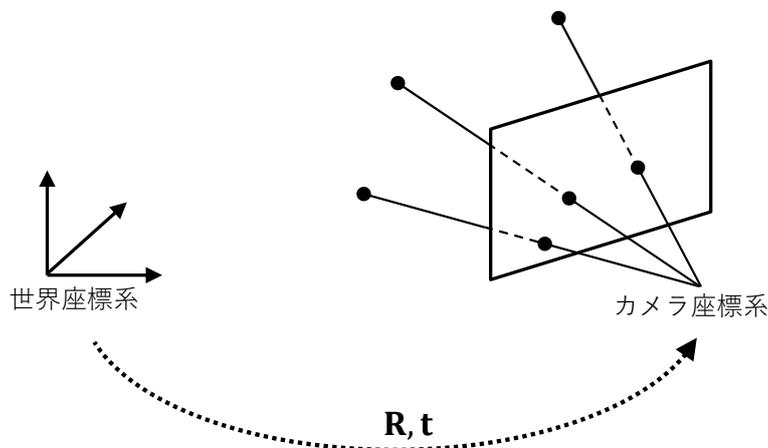


図 1.1: Perspective-n-Point (PnP) 問題

表 1.1: PnP 問題と派生問題における未知パラータと推定に必要な対応点の数

	PnP	PnPf	PnPfr
回転行列 R	✓	✓	✓
並進ベクトル t	✓	✓	✓
焦点距離 f		✓	✓
レンズ歪 k_1, k_2, k_3			✓
未知パラメータ数	6	7	8 (k_1), 10 (k_1, k_2, k_3)
最小点数 n	3	4	4 (k_1), 5 (k_1, k_2, k_3)

PnP 問題への解法

回転行列の表現に依存しない最適性条件を導いたことで、従来不安定とされていた Cayley 変換による回転行列表現でも高速かつ安定な計算を可能にした手法。グレブナー基底を用いて局所解をすべて列挙するため、大域的な最適解が得られることを保証する。また、3次元点の点数や分布に依存しない。

3 問題への統一解法 1

誤差関数を2つの近似した部分問題に分割することで PnP, PnPf, PnPfr 問題を同一のアプローチで解くことを可能にした手法。グレブナー基底を用いることで各問題に対する個別の従来手法と同等の安定性を持ちながら、高速であり点分布への一般性を持つ。点の数に依存しない PnPfr 問題への初めての解法であると同時に、3つの問題への統一的解法を示した初めての手法でもある。

3 問題への統一解法 2

点分布が平面か非平面かによって場合分けをすることで、上記統一解法 1 の高速性をさらに向上させた手法。適切な場合分けにより安定性は失われないことを実験的に示す。

1.3 本論文の構成

本論文は、以下の内容で構成されている。まず、第 2 章では、PnP 問題が登場するアプリケーションにおける役割と解法に要求とされる課題を説明する。それらの課題に対してどのような従来手法が提案され、また、PnPf 問題や PnPfr 問題へと拡張されてきたかという歴史的な経緯を俯瞰する。第 3 章では、PnP 問題に関連が深い研究について紹介する。次に、第 4 章では、ピンホールカメラモデルに基づく射影幾何について解説する。合わせて回転行列の表現方法や誤差関数の定義についても述べる。そして、第 5–7 章にて、新たな解法を提案する。第 5 章では、PnP 問題の最も基礎となる P3P 問題について、特異値分解を要しない新たな直接法を提案する。第 6 章では、グレンナー基底を用いて PnP 問題の大域的な最適解を算出する手法を提案する。第 7 章では、PnP, PnPf, PnPfr 問題の 3 問題を同一のアプローチで解く手法を 2 つ提案する。最後に、第 8 章にて、本論文のまとめを行う。

第 2 章

Perspective-n-Point 問題とその派生問題

本章では、まず PnP 問題が必要とされる場面とその役割について説明する。そして、PnP 問題の解法が解決すべき課題を列挙する。最後に、それらの課題に対してどのような研究が行われてきたか、PnP 問題の歴史的経緯を振り返る。

2.1 PnP 問題の役割

PnP 問題は対応点の数 n に対する拡張性によって大きく 2 つに大別される。1 つは n が固定値で拡張性がない場合であり、例えば P3P 問題や P4Pf 問題が相当する。英語では総称して Minimal problems と呼ばれ、それらに対する解法は Minimal solvers と呼ばれる。定着した日本語訳は存在しないため、本論文では Minimal solvers のことを最小解法と呼ぶ。もう一方の分類は n に対する拡張性があり、 n が可変な場合である。例えば、PnP 問題や PnPf 問題である。これらに対しては Minimal solvers のような呼称は英語でも日本語でも特に定まっていないため、本論文では便宜上拡張解法と呼ぶ。

PnP 問題は画像にコンピュータグラフィックスを合成する Augmented Reality (AR) やリアルタイムに 3 次元地図を作成する Visual Simultaneous Localization And Mapping (VSLAM) にて利用される。AR や VSLAM において、PnP 問題の最小解法と拡張解法は図 2.1 のように使い分けられる。すなわち、最小解法は特徴点マッチングのあとに Random Sample Consensus (RANSAC) [17]^{*1}と組み合わせることで誤対応点を除去するために利用される。RANSAC 実行後にすべての正しい対応点の組を入力として拡張解法を適用し、精度の高いカメラパラメータを得る。このように、最小解法と拡張解法は役割が異なるため両手法の研究が必要となる。具体的なアプリケーション例については第 3 章で紹介する。

^{*1} 全対応点から n 個をランダムサンプリングして最小解法によりパラメータを算出し、そのパラメータと各対応点との誤差を求める。誤差が閾値以上の点を誤った対応点と決定する。このプロセスを反復し、正しい対応点を抽出する。

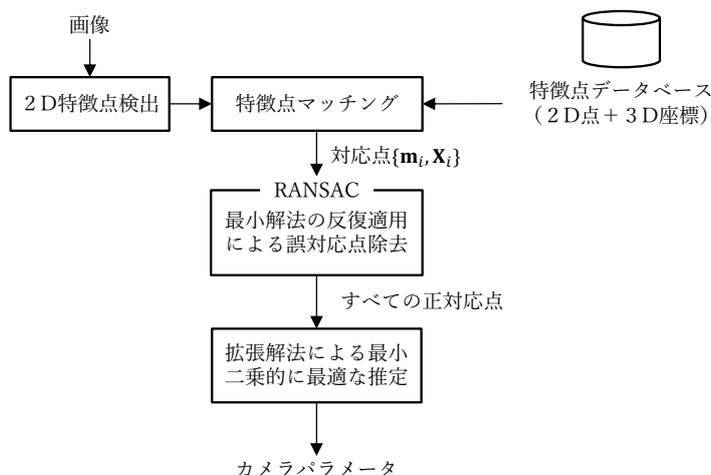


図 2.1: PnP 問題の利用シーン

2.2 PnP 問題の課題

前節で述べたように、PnP 問題はリアルタイム性の高いアプリケーションにおいてカメラ位置推定のサブモジュールとして利用されるため、最小解法も拡張解法も以下の 4 点が課題となる。

- 安定性：数値演算的に高精度で大域的に最適な解を算出できるか
- 高速性：高速に計算可能で実時間アプリケーションに対応できるか
- 一般性：3次元点の分布が平面と非平面のどちらにでも対応できるか
- 拡張性：特定の点数 n に限定した手法か、 n に依らず一貫した手法か

安定性、高速性、一般性の 3 点は最小解法と拡張解法の両方にとって重要な指標である。例えば、理論的には大域的最適解への収束が保証されていても、実装により収束性が大きく左右されるようであれば安定性は高いとは言えない。画像特徴点には一般に観測ノイズが乗るため、ノイズに対する頑健性も含まれる。また、高速であっても誤った解が推定されては実用的ではない。点分布への一般性は、高速性と安定性とのトレードオフなこともある。非平面に限定することで安定性が向上したり、平面に限定することで高速性が向上したりする。例えば、AR の中でも平面マーカを用い場合、平面のみに特化した手法の方が実装上都合が良いことも多い。反対に VSLAM のような点分布の仮定が困難な場合は、手法を切り替えるのは実装が複雑になるため、一般性が高い手法が望ましい。最小解法は基本的に n が固定であるため、拡張性は主に拡張解法にとって重要な課題である。例えば、 n 点から何点かを抽出して組み合わせるような手法であると、組み合わせ爆発により実質的に実行可能な n は限られてしまう。また、拡張解法は最小解法をも内包している場合があり、十分に高速であれば拡張解法と最小解法を使い分ける必要はない。次節では、上記の課題についてどのような研究が行われてきたか、その歴史を振り返る。

2.3 PnP 問題の歴史

1841年に初めてP3P問題への解法[22]が与えられた。以来、射影幾何学や写真測量など異なる研究分野で独立に様々な手法が提案された。これらの手法は導出こそ異なるものの非常に類似しており、1994年にHaralickら[23]により手法間の関係性が整理された。また、1960年代には解が不安定になるDanger Cylinderと呼ばれる縮退条件が知られていたが[72, 78]、解の個数とカメラ配置の幾何的な関係が具体的に明らかにされたのは2003年である[20]。

点の数が $n \geq 3$ となるPnP問題に関しては、3次元点の分布が平面か非平面かに場合分けをしてDirect Linear Transform (DLT)法が適用されていた(DLT法については6.2節を参照)。点の分布に依存しない方式としては、 n 点から任意の3点を選択してP3P問題の解法を繰り返すという手法[5, 62]が2000年代前半までに提案されていたが、組み合わせ爆発により計算量は指数関数的に増大するため実用的とは言えなかった。2000年にLuら[50]は、代数的誤差を用いることで並進ベクトルが回転行列の変数として表されることを示し、並進ベクトルと回転行列を交互に最適化する方法を提案した。提案された交互最適化自体は不適切な局所解に陥ることが多いため主流とはならなかったが、今日よく用いられるPnP問題を回転行列の最適化問題として定式化した初めての論文である。

2000年代後半になると、大域的な最適解への安定性、点の数に依存しない高速性、3次元点の分布に依存しない一般性、それぞれの観点に焦点を当てた研究が行われた。Schweighofer and Pinz [68]は、Luらの手法を平面に限定することで大域的な最適解に収束するように改良した。また、Schweighofer and Pinz [69]は、Luらの定式化を凸最適化の一種であるSum-of-Squares optimizationへと帰着させることで大域的な最適解を保証する手法を提案した。ただし、一般性はなく平面と非平面を区別する必要がある。Moreno-Noguerら[52]は、対応点数の変化に対して計算量の増加が線形時間 $\mathcal{O}(n)$ であるEPnP法を提案した。平面と非平面を区別する必要があったものの、その高速性とコードが公開されたことで広く知られるようになり*2、現在でもベースライン手法として引用されることが多い。点分布を考慮しない一般性と大域的な最適解への安定性を両立したのはHman and Kim [27]である。Schweighofer and Pinz [69]の手法を発展させ、Semidefinite programming relaxation (半正定値計画緩和)により同じく凸最適化でありながら平面も同様に解けることを示した。これら凸最適化のアプローチは理論的には優れているものの、凸最適化に用いる内点法の実装により収束性や計算時間が大きく左右されるという課題があった。

同じく2000年代後半には、2視点幾何におけるグレンナー基底の成功[38, 73, 74]を受けて、焦点距離やレンズ歪みをも未知数とするP4Pf, P4Pfr, P5Pfr問題に対してグレンナー基底を用いる最小解法[10, 11, 29]が提案された。その後、これらの手法はグレン

*2PnP問題への解法として○○PnPという名前が頻繁に使われるきっかけになった。

ナー基底をより効率的に表現するコンパクトなパラメータ化の研究が進んだ [41, 44, 86].

2010 年代に入り, グレブナー基底は PnP 問題にも導入された. Lu ら [50] が定式化した回転行列に関する最適化問題の 1 次最適性条件 (Karush-Kuhn-Tucker 条件) を連立多項式とみなして, 全局所解をグレブナー基底を用いて算出する方法が Hesch ら [26] と中野ら [94, 95] によってほぼ同時期に独立に提案された. 一般性, 高速性, 安定性の両立を謳っていたが, Cayley 変換のスケーリングを無視していたため実際の安定性は高くなかった. そこで, Cayley 変換ではなく四元数を用いる手法 [36, 91] が提案され, グレブナー基底により安定性も達成可能なことが示された. その後, 回転行列の表現によらない KKT 条件が導出され, Cayley 変換でも同様に安定性が実現できることが示された [54]. さらに, 2010 年代後半にかけて PnPf 問題 [90, 93] や PnPfr 問題 [55] に対してもグレブナー基底を利用した手法が提案された.

一方, P3P 問題は 2010 年代まで新しい解法はほとんど議論されることはなかった. しかし, 2011 年に Kneip ら [37] が特異値分解を用いずに外部パラメータを算出する直接解法を提案したことで, 演算量が少なく数値安定性に優れた直接解法が次々に発表された [7, 35, 51, 56, 59]. 中でも, Persson and Nordberg [59] は, P3P 問題を 2 つの円錐曲線の交点とみなすことで 4 次式ではなく 3 次式と 2 次式の組み合わせで解が得られることを示した画期的な手法である.

また, 2010 年代には PnP 問題に新しい問題設定を加えた研究が発表された. 例えば, Inertial Measurement Unit (IMU) を用いて 3 自由度の回転角のうち 2 つの角度を既知とした場合 [82, 39] やローリングシャッターカメラによる画像内の特徴点が時刻同期されない場合 [3, 66], さらに IMU とローリングシャッターカメラを併用した場合 [4] など, スマートフォンの普及によりカメラが IMU を搭載することが一般的になった社会ならではの研究課題である. さらに, 点ではなく線分を用いた Perspective-n-Line (PnL) 問題 [45, 61, 83, 87] も新たに提起され, PnP 問題と同様に安定的な解法について研究されている.

以上, 簡単に PnP 問題とその派生問題の歴史的な経緯を振り返った. 上述したように, PnP 問題は古典的であるにも関わらず現在でも精力的に研究が進められている. 特に演算量を削減するための効率的なパラメータ表現の研究成果は, 多視点幾何を始めとするコンピュータビジョンの様々な幾何学的推定問題にも応用が期待される場所である.

2.4 第 2 章のまとめ

本章では, PnP 問題がリアルタイム性の高い AR や VSLAM などのアプリケーションにおいて利用されることを述べた. その中で点の数 n が固定である最小解法と n が可変の拡張解法の役割の違いについて説明した. 役割の違いとアプリケーションの性質を踏まえて, 実用上では安定性, 高速性, 一般性, 拡張性の 4 点が課題であることを述べた. そして, それらの課題に対する従来研究の歴史的変遷を説明し, 古典的ながら現在でも精力的に研究が行われていることを説明した.

第3章

関連研究

本章では PnP 問題に関連が深い研究について紹介する。

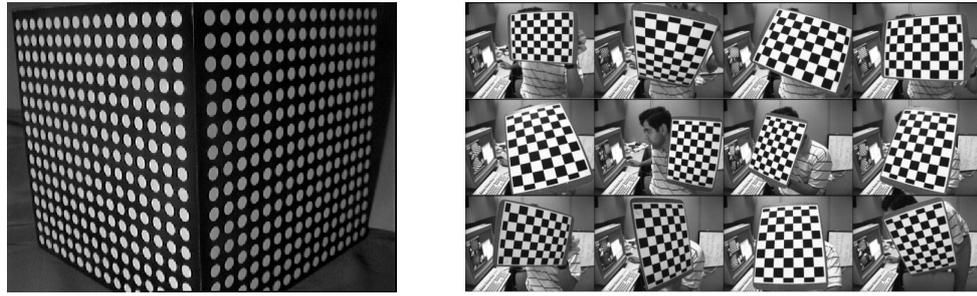
3.1 カメラキャリブレーション

PnP 問題と最も関連が深いのは、校正物体を用いて内部パラメータを算出するカメラキャリブレーションである。カメラキャリブレーションの目的は、3次元座標が既知の校正物体を様々な方向から撮影することで、カメラの内部パラメータを推定することにある。図 3.1 に示すように、キャリブレーション手法は校正物体の形状が非平面 [25, 79] か平面 [89] かの2種類に大別できる。特に、Zhang [89] により提案された平面チェッカーボードを利用する手法は、簡便かつ高精度にパラメータが算出可能なため、今日では標準的な手法として幅広く用いられている。

カメラパラメータを推定するという点では、カメラキャリブレーションと PnP 問題はよく似ている。しかしながら、最も大きな違いは用いられる画像の枚数である。カメラキャリブレーションでは一般に校正物体を撮影した複数枚の画像を用いる。例えば、Zhang の手法では 10 枚以上が推奨とされている。一方、PnP 問題では画像ごとにカメラパラメータが異なるという前提であるため、単一画像のみを用いてパラメータを算出する点が異なる。単にカメラパラメータ推定問題という観点から見ると、PnPf 問題や PnPfr 問題は 1 枚の画像のみを用いたカメラキャリブレーションと捉えることも可能である。

3.2 対応点の検出とマッチング

PnP 問題の前提条件である 3次元座標と画像座標の対応点を得る関連研究を紹介する。第2章で述べたように、PnP 問題では 3次元点と画像特徴点の対応を取ることから始まる。画像特徴点の取得方法としては、Scale Invariant Feature Transform (SIFT) [49] や Speeded-up Robust Features (SURF) [8] を始めとするスケール不変特徴点がよく知られている。携帯端末のようなデバイスでも実行できるように低演算量化を実現した Oriented FAST and rotated BRIEF (ORB) [64] や Fast Retina Keypoint (FREAK) [2]



(a) Heikkilä [25] による直方体を用いる手法 (b) Zhang [89] による平面チェッカーボードを用いる手法 (OpenCV のサンプル画像)

図 3.1: カメラキャリブレーションの例



(a) SIFT [49]

(b) SuperGlue [65]

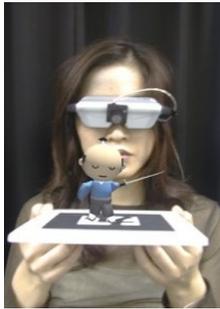
図 3.2: 対応点マッチングの例

なども提案されている。深層学習の発展により畳み込みニューラルネットを用いて視点変化や照明変化への頑健性を向上させる試み [84, 88] も行われている。得られた特徴点をマッチングする手法としては、特徴ベクトルの厳密もしくは近似的な最近傍探索が最も標準的であるが、グラフニューラルネットワークを用いた手法 [65] も提案されている。図 3.2 に特徴点マッチングの具体例を示す。

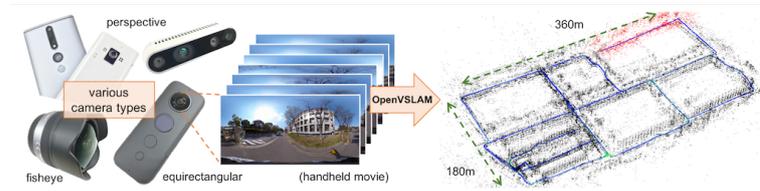
3次元座標を取得する手法としては、LIDAR のようなレンジファインダを用いる手法や、多視点画像から特徴点の3次元座標を復元する Structure-from-Motion (SfM) [24] を用いてもよい。次節にて紹介する Visual-SLAM では、新たな画像が入力されるたびに SfM を反復的に行い、3次元座標を逐次的に推定する。

3.3 アプリケーション

PnP 問題が利用されるアプリケーションとしては、図 3.3 に示すような実写に CG をリアルタイム合成する Augmented Reality (AR) や画像から3次元地図を作成する Visual Simultaneous Localization And Mapping (VSLAM) が挙げられる。正方マーカを用いた低演算なエンジンにより AR を広く知らしめた ARToolkit [34] を始め、現在ではマー



(a) ARToolkit [34]



(b) OpenVSLAM [75]

図 3.3: PnP 問題が登場するアプリケーション例

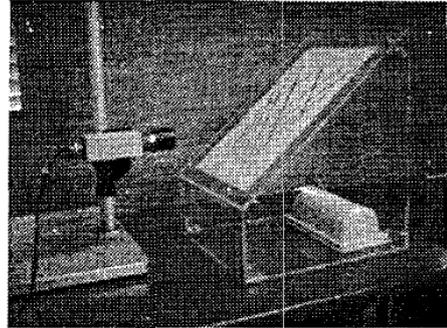
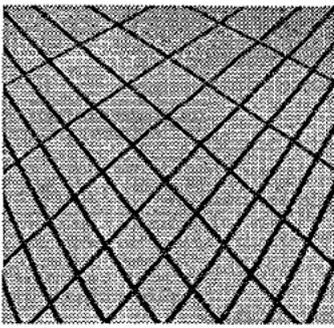


図 3.4: 焦点距離を高精度に推定するための最良格子パターン [33]. 左: パターンの実物. 右: 撮影風景

カーを利用しない Google AR Core^{*1}のようなエンジンも公開されている。VSLAM はロボットナビゲーションのエンジンとして研究が進んでおり、前述した ORB 特徴を利用した ORB-SLAM [53] や OpenVSLAM [75] といった OSS も精力的に研究が行われている。VSLAM においては 3D 点群から撮影環境全体が推測されてしまうプライバシー問題 [60] があり、点群ではなく線分を使ってプライバシー保護を行う研究 [21, 70] が進められている。そのような場面においては点ではなく線分を用いた Perspective-n-Line (PnL) 問題に対する解法 [87, 61, 83, 45] が用いられている。

3.4 ノイズに対する誤差解析

PnP 問題では、3次元座標は正確に計測済みであり、画像平面へ射影される際に観測誤差が加わった特徴点として観測されることを前提としている。画像誤差はカメラのセンサーノイズや量子化誤差、さらに特徴点の検出誤差など様々なノイズの重ね合わせである。そのため、中心極限定理に基づき画像ノイズは平均ゼロの標準正規分布で近似される。

PnP 問題への解法として、Ferraz [16] らは画像特徴点の誤差分布を特徴点のスケール

*1<https://arvr.google.com/arcore/>

を用いて近似し、解の持つ偏差を除去する反復解法を提案している。また、Urban ら [81] は、3次元座標も等方性の標準正規分布に従う誤差を持つと仮定した解法を提案している。3次元座標の誤差分布を推定する方法として、ステレオ画像ペアから推定した3次元座標の誤差共分散行列の理論解析が教科書 [31] に記載されている。カメラキャリブレーションにおいて焦点距離を高精度に算出するために、Kanatani and Maruyama [33] はカメラに対して傾けたときに誤差分散が最小となるような平面のグリッドパターン (図 3.4) を提案している。上述した手法はいずれも1次の誤差解析に基づいているが、Kanatani [32] はノイズが与える偏差を高次の項まで解析し、最尤推定を超える精度が得られることを示している。

3.5 第3章のまとめ

本章では PnP 問題と従来研究との関係についてまとめた。カメラパラメータ推定問題の観点では、PnP 問題は1枚画像を用いたカメラキャリブレーションと解釈できることを指摘した。また、PnP 問題の前提となる対応点の取得方法について説明し、2.1 節で述べたアプリケーションの具体的な研究例を述べた。最後に、PnP 問題だけでなくコンピュータビジョンにおける幾何学的問題に共通する誤差解析の従来研究を紹介した。

第4章

カメラ幾何

本章では、ピンホールカメラモデルに従ったカメラ幾何について説明する。まず、基本的な3次元の画像平面への射影とレンズ歪みについて記述し、本論文における内部パラメータの扱いについて述べる。次に、回転行列を表現する方法のうち、本論文にて登場する表現について説明する。そして、回転行列の制約を満たすように補正する手法と3次元点の分布を推定する手法について解説する。最後に、PnP問題における代表的な2種類の誤差関数について説明する。

4.1 ピンホールカメラモデル

本論文で採用するピンホールカメラモデルについて、図4.1を参照しながら説明する。なお、本論文の座標系はすべて右手系とする。世界座標系におけるある3次元座標 $\mathbf{X} = [x, y, z]^T$ が、解像度 (w, h) の画像平面に射影される関係を考える。カメラは、世界座標系に対して 3×3 の回転行列 \mathbf{R} と 3×1 の並進ベクトル \mathbf{t} により剛体変換されている。また、レンズの水平方向と垂直方向に対する焦点距離をそれぞれ f_u と f_v 、せん断係数^{*1}を γ 、光学中心を (c_u, c_v) とする。3次元座標 \mathbf{X} から画像座標 (u, v) へ射影は、以下の式(4.1)で表される。

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & \gamma & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{bmatrix} \left(\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \right) \quad (4.1)$$

ここで、 r_{ij} は回転行列 \mathbf{R} の (i, j) 成分を表す。式(4.1)の左辺は画像座標であるが、要素1の次元を追加した斉次座標 (homogeneous coordinates) により表現されていることに注意されたい。また、 λ は射影的奥行きと呼ばれる任意の正の実数である。行列ベクトル表記を用いると、式(4.1)は以下の式(4.2)のように簡潔に表すことができる。

$$\lambda \mathbf{m} = \mathbf{K}(\mathbf{R}\mathbf{X} + \mathbf{t}) \quad (4.2)$$

^{*1}四角形から平行四辺形のような、平行性は保つが垂直性を失わせる変形のこと。カメラ幾何においては撮像素子の横軸と縦軸が垂直でないことを表す。

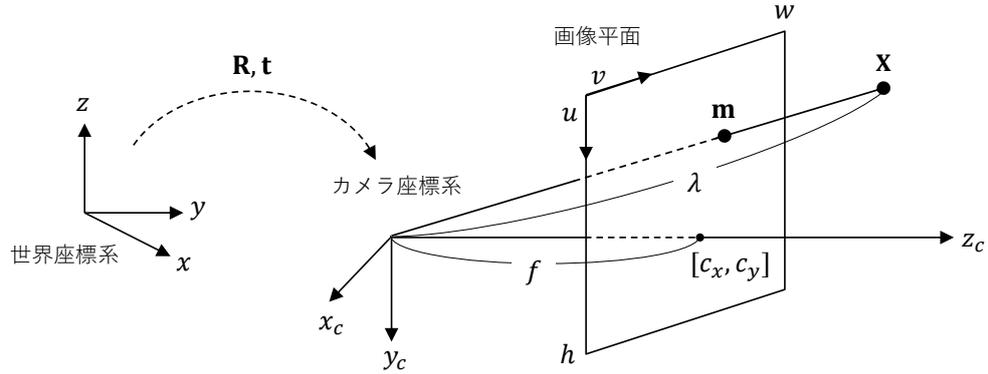


図 4.1: ピンホールカメラモデルにおける射影

ここで,

$$\mathbf{m} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} f_u & \gamma & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{bmatrix} \quad (4.3)$$

である. さらに, 式の両辺が定数倍に等しいことを表す記号 \propto を用いると, 射影的奥行き λ を省略して,

$$\mathbf{m} \propto \mathbf{K}(\mathbf{R}\mathbf{X} + \mathbf{t}) \quad (4.4)$$

と表すこともできる. 画像座標 \mathbf{m} に対して \mathbf{K} の逆行列をかけた座標 $\mathbf{K}^{-1}\mathbf{m}$ を正規化画像座標 (normalized image coordinates) と呼ぶ.

行列 \mathbf{K} と次節にて説明するレンズ歪みは, カメラの動きに影響を受けないため, 総合して内部パラメータ (intrinsic parameters) と呼ばれる. 文献によっては行列 \mathbf{K} のみを指して内部パラメータ行列 (intrinsic matrix) やカメラ校正行列 (camera calibration matrix) と呼ぶこともある. 本論文における内部パラメータの取り扱いについては 4.3 節にて詳細を述べる.

対して, カメラの動きを記述する回転行列 \mathbf{R} と並進ベクトル \mathbf{t} は, 外部パラメータ (extrinsic parameters) と呼ばれる. 日本語では, 回転行列のことを姿勢, 並進ベクトルのことを位置, と呼び, 2つを合わせてカメラの位置姿勢と呼ぶこともある. 英語では, 世界座標系に対する動きであるため absolute camera pose と呼ぶこともある*2. 外部パラメータのうち回転行列は様々な記述方法があるため, 本論文で用いる表現方法について 4.4 節で紹介する.

4.2 レンズ歪み

本論文では, レンズ歪みは半径方向歪み (radial distortion) のみとし, モデル化は division model [13, 18] を用いる. OpenCV などで広く用いられている Brown [9] によ

*2 本論文では扱わないが, あるカメラを座標中心とする複数の画像間のモーションを, 相対的なカメラ位置姿勢 (relative camera pose) と呼ぶ. 詳しくは Hartley and Zisserman [24] を参照のこと

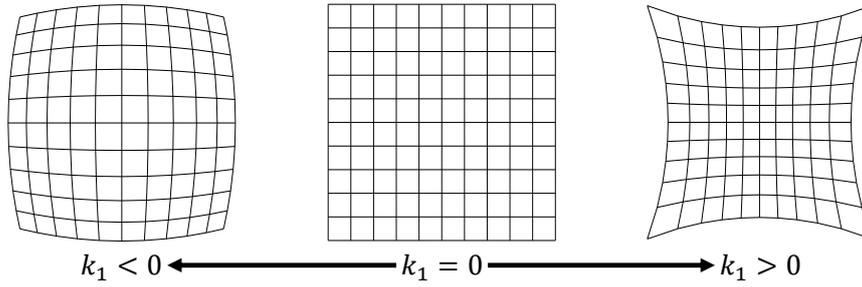


図 4.2: Division model に基づくレンズ歪み

るモデルは、正規化画像座標 $K^{-1}\mathbf{m}$ に歪み係数をかけて歪んだ画像座標を計算する。対に、division model は歪んだ画像座標に歪み係数をかけて歪のない画像座標を計算する。観測値である歪んだ画像座標を用いるため、非線形性が少なく数学的な取り扱いが容易という利点がある。

歪みを含む観測した画像座標を (\tilde{u}, \tilde{v}) 、歪み中心を $(\tilde{c}_u, \tilde{c}_v)$ とすると、division model は次式で表される。

$$\frac{1}{1 + \mathbf{k}^T \mathbf{d}} \begin{bmatrix} \tilde{u} - \tilde{c}_u \\ \tilde{v} - \tilde{c}_v \end{bmatrix} = \begin{bmatrix} u - c_u \\ v - c_v \end{bmatrix} \quad (4.5)$$

ここで、歪み係数 $\mathbf{k} = [k_1, k_2, k_3]^T$ 、 $\mathbf{d} = [d, d^2, d^3]^T$ 、 $d = (\tilde{u} - \tilde{c}_u)^2 + (\tilde{v} - \tilde{c}_v)^2$ である。斉次座標を用いると、式 (4.5) は

$$\begin{bmatrix} \tilde{u} - \tilde{c}_u \\ \tilde{v} - \tilde{c}_v \\ 1 + \mathbf{k}^T \mathbf{d} \end{bmatrix} \propto \begin{bmatrix} u - c_u \\ v - c_v \\ 1 \end{bmatrix} \quad (4.6)$$

と表すことができる。

歪み係数の中でも支配的な k_1 を変化させたときの一例を図 4.2 に示す。望遠レンズにより生じる糸巻き型歪みは $k_1 > 0$ 、広角レンズによる樽型歪みは $k_1 < 0$ で与えられる。OpenCV で用いられている歪みモデルとは正負が逆になることに注意されたい。

4.3 内部パラメータの取り扱い

これまで 4.1~4.2 節にて内部パラメータを詳細に説明してきた。本論文の研究ではそれらすべてを推定することが主眼ではないため、デジタルカメラの利用を前提として次のような近似を行う。

まず、カメラセンサは十分精密に製造されており、縦横の比に偏りが無い ($\gamma = 0$)。また、撮像素子は正方形かつレンズとカメラセンサは正対している ($f_u = f_v = f$)。加えて、レンズ中心とセンサ中心は正確に一致するように装着されている ($c_u = \tilde{c}_u = w/2, c_v = \tilde{c}_v = h/2$)。以上の仮定に基づくと、内部パラメータは焦点距離 f とレンズ歪み \mathbf{k} となる。さらに、レンズ歪みは微小である場合 ($\mathbf{k} \approx \mathbf{0}$)、焦点距離 f のみが未知数である。このとき、事前に $u \leftarrow u - w/2, v \leftarrow v - h/2$ とシフトすれば、内部パラメータ行列は 3×3

表 4.1: 回転表現の比較

	回転行列	オイラー角	四元数	Cayley 変換
変数	9	3	4	3
制約	$\det(\mathbf{R}) = 1$ の直交行列	なし	単位ベクトル	なし
長所	数式表現が簡潔	直感的	計算効率	符号固定
短所	演算誤差の蓄積	ジンバルロック	符号不定性	特異点の存在

の対角行列

$$\mathbf{K} = \begin{bmatrix} f & & \\ & f & \\ & & 1 \end{bmatrix} \propto \begin{bmatrix} 1 & & \\ & 1 & \\ & & 1/f \end{bmatrix} \quad (4.7)$$

として表現される。

上記の仮定は、精密な内部パラメータを要しないアプリケーションであれば十分成立する。これまで PnPf 問題 [10, 93] や PnPfr 問題 [11, 29, 41, 42] および多視点幾何の従来研究 [12, 28, 40, 47, 74] でも広く利用されている。

4.4 回転行列の表現方法

3次元空間における回転行列は自由度3の 3×3 の行列であり、様々な表現方法が存在する。直感的な理解のしやすさや計算の都合に合わせて選択される。本節では代表的な表現方法について説明する。なお、各表現方法の特徴を表4.1にまとめる。

4.4.1 回転行列

右手系における回転行列は行列式が1の直交行列*3

$$\mathbf{R}^T \mathbf{R} = \mathbf{R} \mathbf{R}^T = \mathbf{I}, \quad (4.8)$$

$$\det(\mathbf{R}) = 1 \quad (4.9)$$

である。行列式の制約を表す式(4.9)と等価な表現としては

$$\mathbf{r}_i - \mathbf{r}_j \times \mathbf{r}_k = \mathbf{0}, \quad (i, j, k) = \begin{cases} (1, 2, 3) \\ (2, 3, 1) \\ (3, 1, 2) \end{cases} \quad (4.10)$$

も挙げられる。ここで、 \mathbf{r}_i は \mathbf{R} の第 i 行もしくは第 i 列を指す。

最も単純な表現方法は9個のパラメータ r_{11}, \dots, r_{33} を直接用いることである。他の表現方法と異なり行列成分それぞれが単一のスカラーで表されるため、例えば6.2.1節で後

*3左手系では $\det \mathbf{R} = -1$ となるが、本論文では取り扱わない。

述するように9次元のベクトル化により、数式を簡潔に整理できる利点がある。一方で、 $R_1 R_2 \cdots R_n$ のように複数の回転が乗算されると、数値演算誤差が蓄積して直交制約が次第に満たされなくなる。また、1つの回転行列につき9個の変数をメモリに格納する必要があるため、多数の3Dオブジェクトをアニメーションさせることには向いていない。

4.4.2 ロール・ピッチ・ヨー（オイラー角）

オイラー角による表現の一種であり、次式のように3自由度の回転をz軸（roll角）、y軸（pitch角）、x軸（yaw角）の順に乗算する。

$$\begin{aligned} R &= R_z R_y R_x \\ &= \begin{bmatrix} \cos \theta_z & -\sin \theta_z & 0 \\ \sin \theta_z & \cos \theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta_y & 0 & \sin \theta_y \\ 0 & 1 & 0 \\ -\sin \theta_y & 0 & \cos \theta_y \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_x & -\sin \theta_x \\ 0 & \sin \theta_x & \cos \theta_x \end{bmatrix} \quad (4.11) \end{aligned}$$

各軸の回転を独立に制御できるため直感的に理解しやすく、ロボティクスでよく用いられる。また、加速度センサが利用できると2軸分の角度が既知となるため、残り1軸の回転のみを未知パラメータとして取り扱う際にも便利な表現である [19, 39, 57, 76]。欠点としては、回転行列 R から各角度への分解が一意に決定できず、一般に2つの解が存在することである。また、 $\theta_y = \pm\pi/2$ のときにジンバルロックと呼ばれる3自由度が2自由度に縮退する現象も発生する。ジンバルロックが起きると1つの軸回りの回転しか表現できなくなってしまうため、コンピュータグラフィックスにおいては避けるべき回転表現の1つとされている。

4.4.3 四元数

ジンバルロックが存在せずメモリ効率もよい方法として単位四元数 $\mathbf{q} = [a, b, c, d]^T$ （ただし $\|\mathbf{q}\| = 1$ ）による回転表現がある。2つの回転の内挿補間（Slerp [71]）を簡単に計算でき、L2ノルム正規化で回転行列としての性質が保てるため、コンピュータグラフィックスのアニメーションで広く利用されている。

単位四元数による回転行列は次式で表される。

$$\mathbf{R} = \begin{bmatrix} a^2 + b^2 - c^2 - d^2 & 2(bc - ad) & 2(bd + ac) \\ 2(bc + ad) & a^2 - b^2 + c^2 - d^2 & 2(cd - ab) \\ 2(bd - ac) & 2(cd + ab) & a^2 - b^2 - c^2 + d^2 \end{bmatrix} \quad (4.12)$$

式 (4.12) の各成分がすべて2次の項であることから、 \mathbf{q} と $-\mathbf{q}$ は同じ回転行列を算出することがわかる。単にベクトルの回転を計算する際には正負号を任意に決めればよいが、物体の回転を推定する問題においては正負の不定性が存在するため解空間が広がるという問題がある。

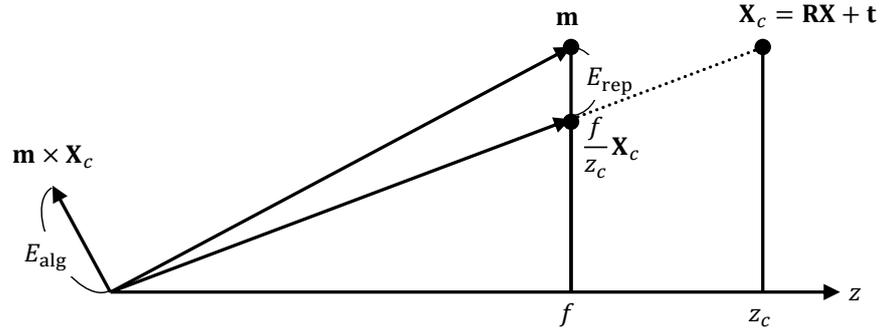


図 4.3: 再投影誤差 E_{rep} と代数的誤差 E_{alg} .

4.4.4 Cayley 変換

Cayley 変換は、 n 次元の歪対称行列と単位行列を用いて n 次元の直交行列を算出する方法である。3 次元の回転行列においては $a = 1$ とおいた四元数 $\mathbf{q} = [1, b, c, d]^T$ に等価である。L2 ノルムが $\|\mathbf{q}\| \neq 1$ であることに注意すると、Cayley 変換による回転行列は

$$\mathbf{R} = \frac{1}{\|\mathbf{q}\|^2} \begin{bmatrix} 1 + b^2 - c^2 - d^2 & 2(bc - d) & 2(bd + c) \\ 2(bc + d) & 1 - b^2 + c^2 - d^2 & 2(cd - b) \\ 2(bd - c) & 2(cd + b) & 1 - b^2 - c^2 + d^2 \end{bmatrix} \quad (4.13)$$

と書ける。符号不定性は存在しないものの、 $a = 0$ となる四元数を表現できないため、回転の推定問題においては特異点となってしまふ。それを避けるためには、事前に $a \neq 0$ であるランダムな回転行列をかけるなどの前処理が必要となる。

4.5 回転行列の補正

何らかの方法（例えば 6.2.1 節に登場する Direct Linear Transform (DLT) 法など）で算出した回転行列は、式 (4.8)–(4.9) に示す直交制約と行列式の制約を厳密には満たさないことがある。そこで、それらの制約を満たすように補正しなくてはならない。回転行列の補正は閉形式 (closed-form) で求められることが知られている [80]。

ある $n \times n$ の行列 A に対してフロベニウス距離が最も近い回転行列 R は、特異値分解 $A = UDV^T$ を用いると

$$R = U \text{diag}(1, 1, \det(UV^T)) V^T \quad (4.14)$$

として算出できる。

4.6 平面/非平面の判定

AR マーカのように 3 次元点 \mathbf{X}_i の分布が既知である場合を除き、カメラ位置推定手法が一般性を持たない場合は \mathbf{X}_i が平面上の点か否かを判定する必要がある。点分布は、

2次モーメント行列の階数を調べれば以下のように判定できる。

$$\text{rank} \left(\sum_{i=1}^n (\mathbf{X}_i - \bar{\mathbf{X}})(\mathbf{X}_i - \bar{\mathbf{X}})^T \right) = \begin{cases} 3 & \text{非平面} \\ 3 \text{ かつ条件数が大きい} & \text{准平面} \\ 2 & \text{平面} \\ 1 & \text{直線} \end{cases} \quad (4.15)$$

ただし、 $\bar{\mathbf{X}} = 1/n \sum_{i=1}^n \mathbf{X}_i$ は点群の重心である。

2次モーメント行列が実対称行列のため、式 (4.15) における条件数とは最大特異値を最小特異値で割った値である。平面であれば階数が2なので最小特異値はゼロであり、条件数は無限大となる。ほぼ平面の場合は、無限大にはならないが階数が3、ということである。「ほぼ平面」を平面の解法か非平面の解法かに分類しなくてはならないため、一般性を持たない方式にとって条件数のしきい値設定は重要なパラメータである。

4.7 誤差関数

コンピュータビジョンにおけるカメラ位置姿勢推定問題における誤差関数は、一般に2種類に大別できる。ひとつは再投影誤差 (reprojection error) と呼ばれる物理的に意味のある指標であり、もうひとつは代数的誤差 (algebraic error) と呼ばれる数学的な指標である。図 4.3 にそれぞれの誤差関数の概要を示す。

再投影誤差とは、観測した画像座標と画像平面へ射影した3次元座標とのユークリッド距離のことである。カメラ座標系における3次元座標を $\mathbf{X}_c = \mathbf{R}\mathbf{X} + \mathbf{t}$ とすると、PnP問題における再投影誤差は

$$E_{\text{rep}} = \left\| \mathbf{m} - \frac{1}{z_c} \mathbf{K}\mathbf{X}_c \right\|^2 \quad (4.16)$$

と書ける。ここで、 $z_c = \mathbf{r}_3^T \mathbf{X} + t_3$ である。式 (4.16) は $1/z_c$ の存在により非線形関数であるため、再投影誤差の最小化は一般的には Levenberg Marquard 法のような反復解法により行われる。最も良い精度が得られる方法と考えられているが、十分良い初期値を与えないと不正確な局所解に陥ってしまうリスクがある。加えて、前節で述べた回転行列の制約条件を満たしながら反復解法を行う必要がある。例えば、オイラー角を直接に最適化する方法 [34] や、 $\|\mathbf{q}\| = 1$ を保ちながら四元数の勾配を記述する方法 [67, 77] や、微小回転を線形化するリー代数による方法 [85, 97] などが挙げられる。C++ 言語の最適化ツールである Ceres Solver [1] には回転行列の偏微分関数が実装されている。

代数的誤差とは、ベクトルの平行性に着目した指標である。PnP問題において、斉次化した画像座標 \mathbf{m} とカメラ座標系における3次元座標 \mathbf{X}_c は、一切のノイズがなければその方向は完全に一致するはずである。そこで、ベクトルのクロス積を取れば、次式のように射影的奥行きを用いずに誤差を表現できる。

$$E_{\text{alg}} = \|\mathbf{m} \times (\mathbf{K}\mathbf{X}_c)\|^2 \quad (4.17)$$

上式は、式 (4.4) の両辺のクロス積に等しい。図 4.3 に示すように、代数的誤差は \mathbf{m} と

KX_c に直交するベクトルの長さに相当するが、2つのベクトルのスケーリング次第でいくらでも大きさは変動する。それゆえ、代数的誤差はなんら物理的な意味を持たない。しかしながら、画像ノイズが小さい場合には再投影誤差最小化に匹敵する精度を持つことが実験的に示されてきた [24]。また、画像ノイズが大きい場合であっても、再投影誤差最小化の良い初期値を与える方法として標準的に用いられている。

4.8 第4章のまとめ

本章では PnP 問題の理解に必要なカメラ幾何と周辺の数学的な内容について解説した。中でも回転表現の取り扱いは、従来手法や本論文における提案手法の導出の違いにも深く関係しているため、各手法を特徴づける重要な要素である。また、平面と非平面の判定方法は Direct Linear Transform (DLT) 法において重要な前処理の一つである。本論文ではもっぱら代数的誤差を扱うが、一般に再投影誤差の最小化が最適と言われているため、それぞれの特性を理解して適宜使い分けることが必要となる。

第 5 章

P3P 問題への提案手法

本章では P3P 問題について議論する。P3P 問題は、 $n = 3$ とした PnP 問題の最も基礎的な形式である。まず、P3P 問題の定式化を行い、次に、代表的な従来手法を簡単に説明する。従来手法は、外部パラメータの算出に特異値分解を用いる非直接解法と、特異値分解を用いない直接解法の 2 種類に分類される。そして、直接解法に属する提案手法の詳細な導出を行い、シミュレーション実験にて従来手法との性能を比較する。

5.1 P3P 問題の定式化

P3P 問題において内部パラメータは既知であるから、式 (4.2) において $\mathbf{m} \leftarrow K^{-1}\mathbf{m}$ と正規化画像座標へ置き換えると、以下の連立方程式として定式化できる。

$$\begin{aligned}\lambda_1 \mathbf{m}_1 &= R\mathbf{X}_1 + \mathbf{t} \\ \lambda_2 \mathbf{m}_2 &= R\mathbf{X}_2 + \mathbf{t} \\ \lambda_3 \mathbf{m}_3 &= R\mathbf{X}_3 + \mathbf{t}\end{aligned}\tag{5.1}$$

これを解いて回転行列 R と並進ベクトル \mathbf{t} を求めるのが P3P 問題である。なお、表記を簡単にするため、本章ではそれぞれの正規化画像座標の L2 ノルムを $\|\mathbf{m}\| = 1$ となるように設定する*1。

5.2 従来手法

5.2.1 非直接解法

古典的な非直接解法 [20, 23] では、まず射影的奥行き λ に関する 4 次式を導出し、回転行列と並進ベクトルを特異値分解により算出する。式 (5.1) から 2 式のペア $\{i, j\}$ を選び、差を取って並進ベクトル \mathbf{t} を消去する。

$$\lambda_i \mathbf{m}_i - \lambda_j \mathbf{m}_j = R(\mathbf{X}_i - \mathbf{X}_j)\tag{5.2}$$

*1任意のスケール s に対して $\lambda \mathbf{m} = (\lambda/s)(s\mathbf{m})$ が成り立つため、 $\lambda \leftarrow \lambda/s$ と読み替えればよい。

次に、回転行列 \mathbf{R} の直交性 $\mathbf{R}^T \mathbf{R} = \mathbf{I}$ を利用して回転行列 \mathbf{R} を消去する. 式 (5.2) の両辺の内積を取ると, \mathbf{X}_i と \mathbf{X}_j の距離 d_{ij} を算出できる.

$$\lambda_i^2 - 2\lambda_i \lambda_j \mathbf{m}_i^T \mathbf{m}_j + \lambda_j^2 = d_{ij}^2 \quad (5.3)$$

ここで, $d_{ij}^2 = \|\mathbf{X}_i - \mathbf{X}_j\|^2$ である. また, 5.1 節で述べたように $\|\mathbf{m}_i\| = 1$ と正規化していることに注意されたい. 式 (5.3) は 3 組のペア $\{1, 2\}, \{2, 3\}, \{3, 1\}$ について成り立つので, 2 式ずつ取り出すと次式を得る.

$$\begin{aligned} d_{23} (\lambda_1^2 - 2\lambda_1 \lambda_2 \mathbf{m}_1^T \mathbf{m}_2 + \lambda_2^2) - d_{12} (\lambda_2^2 - 2\lambda_2 \lambda_3 \mathbf{m}_2^T \mathbf{m}_3 + \lambda_3^2) &= 0 \\ d_{31} (\lambda_1^2 - 2\lambda_1 \lambda_2 \mathbf{m}_1^T \mathbf{m}_2 + \lambda_2^2) - d_{12} (\lambda_3^2 - 2\lambda_3 \lambda_1 \mathbf{m}_3^T \mathbf{m}_1 + \lambda_1^2) &= 0 \end{aligned} \quad (5.4)$$

式 (5.4) は, 任意の λ_i について定数倍しても解は変わらない. そこで, $\lambda_1 = 1, x = \lambda_2/\lambda_1, y = \lambda_3/\lambda_1$ とおくと, 以下の 2 変数連立方程式として書き換えられる.

$$\begin{aligned} f(x, y) &= f_1 x^2 + f_2 xy + f_3 y^2 + f_4 x + f_6 = 0 \\ g(x, y) &= g_1 x^2 + g_3 y^2 + g_4 x + g_5 y + g_6 = 0 \end{aligned} \quad (5.5)$$

ただし,

$$\begin{aligned} f_1 &= d_{23} - d_{12}, \quad f_2 = 2d_{12} \mathbf{m}_2^T \mathbf{m}_3, \quad f_3 = -2d_{23} \mathbf{m}_1^T \mathbf{m}_2, \quad f_4 = -d_{12}, \quad f_6 = d_{23} \\ g_1 &= d_{31}, \quad g_3 = -2d_{31} \mathbf{m}_1^T \mathbf{m}_2, \quad g_4 = -d_{12}, \quad g_5 = 2d_{12} \mathbf{m}_3^T \mathbf{m}_1, \quad g_6 = d_{31} - d_{12} \end{aligned} \quad (5.6)$$

である. さらに, $f(x, y)$ と $g(x, y)$ から y を消去すると x についての 4 次式を得る.

$$h(x) = h_1 x^4 + h_2 x^3 + h_3 x^2 + h_4 x + h_5 = 0 \quad (5.7)$$

ここで,

$$\begin{aligned} h_1 &= f_1^2 g_3^2 - 2f_1 f_3 g_1 g_3 + f_2^2 g_1 g_3 + f_3^2 g_1^2 \\ h_2 &= 2f_1 f_4 g_3^2 + 2f_3^2 g_1 g_4 + f_2^2 g_3 g_4 - f_1 f_2 g_3 g_5 - 2f_1 f_3 g_3 g_4 - f_2 f_3 g_1 g_5 - 2f_3 f_4 g_1 g_3 \\ h_3 &= f_3^2 g_4^2 + f_4^2 g_3^2 + f_1 f_3 g_5^2 + 2f_1 f_6 g_3^2 + 2f_3^2 g_1 g_6 + f_2^2 g_3 g_6 - 2f_1 f_3 g_3 g_6 - 2f_3 f_6 g_1 g_3 \\ &\quad - f_2 f_3 g_4 g_5 - f_2 f_4 g_3 g_5 - 2f_3 f_4 g_3 g_4 \\ h_4 &= f_3 f_4 g_5^2 + 2f_4 f_6 g_3^2 + 2f_3^2 g_4 g_6 - f_2 f_3 g_5 g_6 - f_2 f_6 g_3 g_5 - 2f_3 f_4 g_3 g_6 - 2f_3 f_6 g_3 g_4 \\ h_5 &= f_3^2 g_6^2 - 2f_3 f_6 g_3 g_6 + f_3 f_6 g_5^2 + f_6^2 g_3^2 \end{aligned} \quad (5.8)$$

である. 式 5.7 は Ferrari の公式により閉形式で x の解を求めることができる. 4 次式であるから, 最大で 4 個の実数解が存在する. x の実数解を式 (5.5) に代入すれば対応する y が得られる. そして, 式 (5.3) に基づいて x と y から $\lambda_1, \lambda_2, \lambda_3$ を復元する.

得られた射影的奥行き $\lambda_1, \lambda_2, \lambda_3$ を用いて, 回転行列 \mathbf{R} と並進ベクトル \mathbf{t} を算出する. 式 (5.1) に射影的奥行きを代入すると, 回転行列 \mathbf{R} と並進ベクトル \mathbf{t} について以下の最適化問題を定式化できる.

$$\begin{aligned} \min_{\mathbf{R}, \mathbf{t}} \quad & \sum_{i=1}^3 \|\lambda_i \mathbf{m}_i - (\mathbf{R} \mathbf{X}_i + \mathbf{t})\|^2 \\ \text{s.t.} \quad & \mathbf{R}^T \mathbf{R} = \mathbf{I}, \quad \det(\mathbf{R}) = 1 \end{aligned} \quad (5.9)$$

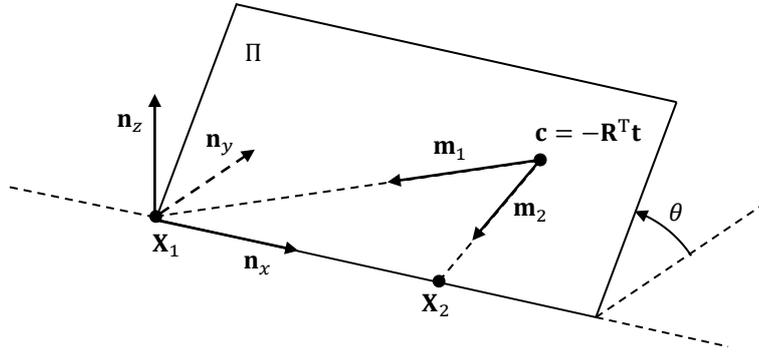


図 5.1: Kneip らによる P3P 解法 [37] の図解

式 (5.9) の解は以下で与えられる.

$$\mathbf{R} = \mathbf{U} \text{diag}(1, 1, \det(\mathbf{UV}^T)) \mathbf{V}^T, \quad \mathbf{t} = \bar{\mathbf{m}} - \mathbf{R}\bar{\mathbf{X}} \quad (5.10)$$

ここで,

$$\bar{\mathbf{m}} = \frac{1}{3} \sum_{i=1}^3 \lambda_i \mathbf{m}_i, \quad \bar{\mathbf{X}} = \frac{1}{3} \sum_{i=1}^3 \mathbf{X}_i, \quad \mathbf{UDV}^T = \sum_{i=1}^3 (\mathbf{X}_i - \bar{\mathbf{X}}) (\lambda_i \mathbf{m}_i - \bar{\mathbf{m}})^T \quad (5.11)$$

である. 詳細な導出は文献 [15, 80] を参照されたい.

上述した古典的な非直接解法の課題は, 式 (5.9) を解く際に特異値分解を用いるため, 処理時間がかかるにも関わらず最終的な回転行列 \mathbf{R} と並進ベクトル \mathbf{t} の演算精度が失われることである. そのため, 次節で紹介する特異値分解を用いない直接解法が提案された.

5.2.2 直接解法

Kneip らの手法 [37]

最初の直接解法は Kneip らにより 2011 年に提案された. Kneip らの手法の骨子は, 幾何的な解析により 1 つの角度に関する 4 次式を導出することである. 具体的には, 図 5.1 に示すように, カメラの 3 次元位置 $\mathbf{c} = -\mathbf{R}^T \mathbf{t}$ と \mathbf{X}_1 と \mathbf{X}_2 が作る平面 Π とベクトル \mathbf{n}_y が作る角度 θ を未知数とする. ここで,

$$\begin{aligned} \mathbf{n}_x &= \frac{\mathbf{X}_2 - \mathbf{X}_1}{\|\mathbf{X}_2 - \mathbf{X}_1\|} \\ \mathbf{n}_z &= \frac{\mathbf{n}_x \times (\mathbf{X}_3 - \mathbf{X}_1)}{\|\mathbf{n}_x \times (\mathbf{X}_3 - \mathbf{X}_1)\|} \\ \mathbf{n}_y &= \mathbf{n}_z \times \mathbf{n}_x \end{aligned} \quad (5.12)$$

である. θ に関する 4 次式の導出は複雑なため本論文では省略する. その複雑さゆえ, その後の直接解法の発展につながったとも言える.

Ke らの手法 [35]

Ke らも幾何的な解析に基づき 1 つの角度に関する 4 次式を導出した。その導出は Kneip 以上に複雑なため、同様に本論文では省略する。

Banno の手法 [7]

Banno は回転行列 R と並進ベクトル t を射影的奥行き $\lambda_1, \lambda_2, \lambda_3$ の線形結合で表せることを発見した。式 (5.1) は以下のように書き換えられる。

$$\begin{bmatrix} \mathbf{S}_1 & \mathbf{I} & -\mathbf{m}_1 & & \\ \mathbf{S}_2 & \mathbf{I} & & -\mathbf{m}_2 & \\ \mathbf{S}_3 & \mathbf{I} & & & -\mathbf{m}_3 \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \mathbf{t} \\ \boldsymbol{\lambda} \end{bmatrix} = \mathbf{A}\mathbf{p} = \mathbf{0} \quad (5.13)$$

ここで、 \mathbf{r} は R の 9 次元ベクトル表記であり、

$$\mathbf{S}_i = \begin{bmatrix} \mathbf{X}_i^T & & \\ & \mathbf{X}_i^T & \\ & & \mathbf{X}_i^T \end{bmatrix}, \quad \boldsymbol{\lambda} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} \quad (5.14)$$

である。式 (5.13) より、 \mathbf{p} は \mathbf{A} のゼロ空間ベクトルの線形結合で表される。すなわち、

$$\mathbf{p} \propto \text{span}(\text{null}(\mathbf{A})) \quad (5.15)$$

であり、しかも、その係数は $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \lambda_3]^T$ に一致する。回転行列の直交性を利用すると $\boldsymbol{\lambda}$ についての 4 次式を得られ、その実数解を式 (5.15) に代入すると回転行列 R と並進ベクトル t が決定できる。

Persson らの手法 [59]

Persson and Nordberg は 4 次式を使わずに P3P 問題が解けることを初めて示した。その骨子は、式 (5.4) を 2 つの円錐曲線の交点として解くことである。式 (5.4) を 2 次形式で表すと、

$$\boldsymbol{\lambda}^T \mathbf{C}_1 \boldsymbol{\lambda} = 0, \quad \boldsymbol{\lambda}^T \mathbf{C}_2 \boldsymbol{\lambda} = 0, \quad (5.16)$$

と書き換えられる。ここで、 \mathbf{C}_1 と \mathbf{C}_2 は 3×3 の対称行列である。円錐曲線の交点の算出は [63] に詳細が記載されているため、ここでは手順の概略のみ記す。まず、2 つの円錐曲線の交点を通る直線を見つけるため、次式を解いて退化した円錐曲線を算出する。

$$\det(\mathbf{C}_1 + \gamma \mathbf{C}_2) = 0. \quad (5.17)$$

上式は γ に関する 3 次式である。そして、退化した円錐曲線は 2 本の直線 \mathbf{p}, \mathbf{q} を用いて

$$\mathbf{C}_1 + \gamma \mathbf{C}_2 = \mathbf{p}\mathbf{q}^T + \mathbf{q}\mathbf{p}^T \quad (5.18)$$

と表せる。2 本の直線は \mathbf{C}_1 の一点を通るので、その交点について次式が成り立つ

$$\begin{cases} \boldsymbol{\lambda}^T \mathbf{C}_1 \boldsymbol{\lambda} = 0 \\ \mathbf{p}^T \boldsymbol{\lambda} = 0 \end{cases}, \quad \begin{cases} \boldsymbol{\lambda}^T \mathbf{C}_1 \boldsymbol{\lambda} = 0 \\ \mathbf{q}^T \boldsymbol{\lambda} = 0 \end{cases} \quad (5.19)$$

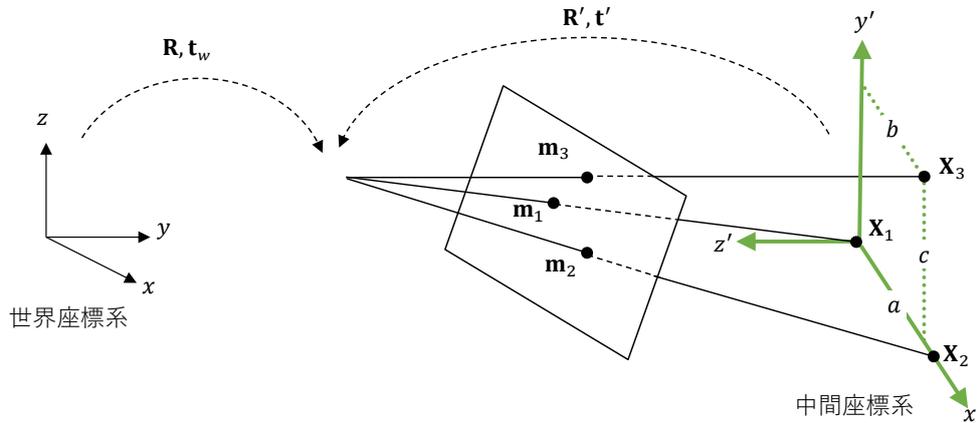


図 5.2: 提案手法における中間座標系

式 (5.19) は λ の定数倍に不定であるから、非直接解法と同様に λ の比を用いれば 2 次式で表せる。

さらに、Persson and Nordberg は射影的奥行きを利用して、特異値分解を用いずに回転行列と並進ベクトルを算出する方法を示している。式 (5.2) より、3 組の組み合わせから 2 組を選ぶと次式を得る。

$$[\mathbf{y}_{12}, \mathbf{y}_{23}, \mathbf{y}_{12} \times \mathbf{y}_{23}] = \mathbf{R} [\mathbf{z}_{12}, \mathbf{z}_{23}, \mathbf{z}_{12} \times \mathbf{z}_{23}] \quad (5.20)$$

ここで、 $\mathbf{y}_{ij} = \lambda_i \mathbf{m}_i - \lambda_j \mathbf{m}_j$, $\mathbf{z}_{ij} = \mathbf{X}_i - \mathbf{X}_j$ である。上式の右から $[\mathbf{z}_{12}, \mathbf{z}_{23}, \mathbf{z}_{12} \times \mathbf{z}_{23}]^{-1}$ をかければ \mathbf{R} が得られる。そして、3 組の対応点のうち任意の 1 点について $\mathbf{t} = \mathbf{R}\mathbf{X}_i - \lambda_i \mathbf{m}_i$ として並進ベクトルが求まる。

5.3 提案手法

本節では文献 [56] をもとに新たな直接解法について説明する。

5.3.1 中間座標系への変換

P3P 問題は、すべての 3 次元座標が $z = 0$ 上の平面に存在すると考えても一般性を失わない。そこで、図 5.2 に示すように、入力された 3 次元座標を $z = 0$ の中間座標系へと変換する*2。1 番目の 3 次元座標 \mathbf{X}_1 を中心とすると、中間座標系における 3 次元座標 \mathbf{X}' は以下のように計算できる。

$$\mathbf{X}'_i = \mathbf{N}^T (\mathbf{X}_i - \mathbf{X}_1) \rightarrow \mathbf{X}'_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{X}'_2 = \begin{bmatrix} a \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{X}'_3 = \begin{bmatrix} b \\ c \\ 0 \end{bmatrix}, \quad (5.21)$$

*2 この仮定は Kneip [37] や Banno [7] と同様である。

ここで、式 (5.12) を用いて $N = [\mathbf{n}_x, \mathbf{n}_y, \mathbf{n}_z]$ であり、

$$a = \|\mathbf{X}_2 - \mathbf{X}_1\|, \quad c > 0, \quad \sqrt{b^2 + c^2} = \|\mathbf{X}_3 - \mathbf{X}_1\| \quad (5.22)$$

である。行列 N は直交行列 ($NN^T = N^T N = I$) であるから、世界座標系と中間座標系の剛体変換は以下のように書ける。

$$\begin{aligned} R\mathbf{X}_i + \mathbf{t} &= (RN) (N^T(\mathbf{X}_i - \mathbf{X}_1)) + (\mathbf{t} + R\mathbf{X}_1) \\ &= R'\mathbf{X}'_i + \mathbf{t}'. \end{aligned} \quad (5.23)$$

よって、中間座標系における新しい回転行列 R' と並進ベクトル \mathbf{t}' は

$$\begin{aligned} R' &= RN, \\ \mathbf{t}' &= \mathbf{t} + R\mathbf{X}_1 \end{aligned} \quad (5.24)$$

と表せる。そして、式 (5.21) と式 (5.24) を用いると、中間座標系における P3P 問題は次式のように書ける。

$$\lambda_1 \mathbf{m}_1 = \mathbf{t}', \quad (5.25)$$

$$\lambda_2 \mathbf{m}_2 = a \mathbf{r}_1 + \mathbf{t}', \quad (5.26)$$

$$\lambda_3 \mathbf{m}_3 = b \mathbf{r}_1 + c \mathbf{r}_2 + \mathbf{t}', \quad (5.27)$$

ここで、 \mathbf{r}_1 と \mathbf{r}_2 は、それぞれ R' の第1列目と第2列目を指す。

5.3.2 射影的奥行きの算出

式 (5.25) から式 (5.26) を引いて \mathbf{t}' を消去すると、 \mathbf{r}_1 について次式を得る。

$$\begin{aligned} \mathbf{r}_1 &= \frac{1}{a} (\lambda_2 \mathbf{m}_2 - \lambda_1 \mathbf{m}_1) \\ &= \frac{1}{a} \mathbf{A} \boldsymbol{\lambda}, \end{aligned} \quad (5.28)$$

ここで、 $\mathbf{A} = [-\mathbf{m}_1, \mathbf{m}_2, \mathbf{0}]$, $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \lambda_3]^T$ である。同様に、式 (5.25) から式 (5.27) を引き、さら式 (5.28) を代入すると、 \mathbf{r}_2 について次式を得る。

$$\begin{aligned} \mathbf{r}_2 &= \frac{1}{c} (\lambda_3 \mathbf{m}_3 - \lambda_1 \mathbf{m}_1 - b \mathbf{r}_1) \\ &= \frac{1}{c} (\mathbf{B} - p \mathbf{A}) \boldsymbol{\lambda} \\ &= \frac{1}{c} \mathbf{C} \boldsymbol{\lambda}, \end{aligned} \quad (5.29)$$

ここで、 $p = b/a$, $\mathbf{B} = [-\mathbf{m}_1, \mathbf{0}, \mathbf{m}_3]$, $\mathbf{C} = \mathbf{B} - p \mathbf{A}$ である。

次に、射影的奥行き $\boldsymbol{\lambda}$ に関する連立方程式を立てる。 \mathbf{r}_1 と \mathbf{r}_2 は直交行列の第1列、第2列であるから、

$$\mathbf{r}_1^T \mathbf{r}_2 = 0, \quad (5.30)$$

$$\mathbf{r}_1^T \mathbf{r}_1 - \mathbf{r}_2^T \mathbf{r}_2 = 0. \quad (5.31)$$

が成り立つ。上記の式 (5.30) と式 (5.31) はともに定数倍に不定であるので、新たに2つの変数 $x = \lambda_2/\lambda_1$, $y = \lambda_3/\lambda_1$ を導入して未知数を削減する。そして $\boldsymbol{\mu} = \boldsymbol{\lambda}/\lambda_1 = [1, x, y]^T$ と置き換えると、式 (5.30) と式 (5.31) は以下のように書き換えられる。

$$\mathbf{r}_1^T \mathbf{r}_2 = \frac{1}{ac} \boldsymbol{\lambda}^T \mathbf{A}^T \mathbf{C} \boldsymbol{\lambda} \propto \boldsymbol{\mu}^T (p \mathbf{A}^T \mathbf{A} - \mathbf{A}^T \mathbf{B}) \boldsymbol{\mu}, \quad (5.32)$$

$$\mathbf{r}_1^T \mathbf{r}_1 - \mathbf{r}_2^T \mathbf{r}_2 = \frac{1}{a^2} \boldsymbol{\lambda}^T \mathbf{A}^T \mathbf{A} \boldsymbol{\lambda} - \frac{1}{c^2} \boldsymbol{\lambda}^T \mathbf{C}^T \mathbf{C} \boldsymbol{\lambda} \propto \boldsymbol{\mu}^T (q \mathbf{A}^T \mathbf{A} - \mathbf{B}^T \mathbf{B}) \boldsymbol{\mu}, \quad (5.33)$$

ここで、 $q = (b^2 + c^2)/a^2$ である。式 (5.32) と式 (5.33) を展開すると、 x と y についての2変数2次方程式を得る。

$$\begin{aligned} f(x, y) &= f_1 x^2 + f_2 xy + f_4 x + f_5 y + f_6 = 0, \\ g(x, y) &= g_1 x^2 + g_3 y^2 + g_4 x + g_5 y + g_6 = 0, \end{aligned} \quad (5.34)$$

各方程式の係数は以下の通りである*³。

$$\begin{aligned} f_1 &= p, & f_2 &= -\mathbf{m}_2^T \mathbf{m}_3, & f_4 &= (1 - 2p)(\mathbf{m}_1^T \mathbf{m}_2), & f_5 &= \mathbf{m}_1^T \mathbf{m}_3, & f_6 &= p - 1, \\ g_1 &= q, & g_3 &= -1, & g_4 &= -2q(\mathbf{m}_1^T \mathbf{m}_2), & g_5 &= 2f_5, & g_6 &= q - 1. \end{aligned} \quad (5.35)$$

式 (5.32)–(5.35) の詳細な導出は章末の付録 5.6 を参照のこと。

$f(x, y)$ において y^2 の係数はゼロであるから、 y は x の関数として表せる。

$$y = -\frac{f_1 x^2 + f_4 x + f_6}{f_2 x + f_5}. \quad (5.36)$$

式 (5.36) を $g(x, y)$ に代入すると、以下のように x に関する4次式を得る。

$$h(x) = h_1 x^4 + h_2 x^3 + h_3 x^2 + h_4 x + h_5, \quad (5.37)$$

ここで、

$$\begin{aligned} h_1 &= f_2^2 g_1 - f_1^2, \\ h_2 &= f_2^2 g_4 + 2f_2 f_5 (g_1 - f_1) - 2f_1 f_4, \\ h_3 &= f_5^2 (g_1 - 2f_1) + 2f_2 f_5 (g_4 - f_4) - 2f_1 f_6 + f_2^2 g_6 - f_4^2, \\ h_4 &= f_5^2 (g_4 - 2f_4) + 2f_2 f_5 (g_6 - f_6) - 2f_4 f_6, \\ h_5 &= f_5^2 (g_6 - 2f_6) - f_6^2. \end{aligned} \quad (5.38)$$

である。

非直接解法と同様に、4次式の解は Ferrari の公式や Descartes の公式を用いて閉形式で求めることができる。結果として、式 (5.37) から最大4個の x の実数解が得られる。そして、得られた x を式 (5.36) に代入して対応する y が得られる。加えて、閉形式で得られた解の数値演算精度を向上するためにニュートン法を任意に適用してもよい。この操作を英語では Root polishing と呼ぶが、日本語では定着した訳語がないため本論文ではそのまま英語の呼び名を使用する。本論文では、式 (5.34) における x と y に対してニュートン法を1回だけ適用した。Root polishing により生じる処理時間の増加については 5.4.3 節にて議論する。

*³文献 [56] では f_2 の符号が誤っており、マイナス記号が抜けている。

Algorithm 1 A simple direct P3P solver**Input:** \mathbf{X}_i : 3D points, \mathbf{m}_i : normalized 2D points ($\|\mathbf{m}_i\| = 1$)**Output:** \mathbf{R} : rotation matrix, \mathbf{t} : translation vector

- 1: compute \mathbf{N} and $\mathbf{X}'_i \leftarrow \mathbf{N}^\top(\mathbf{X}_i - \mathbf{X}_1)$ ▷ Eq.(5.12)
- 2: compute $p, q, \mathbf{A}, \mathbf{B}, \mathbf{C}$ ▷ Eqs.(5.28) and (5.29)
- 3: compute coefficients of $f(x, y), g(x, y), h(x)$ ▷ Eqs.(5.35) and (5.38)
- 4: find roots of $h(x)$
- 5: **for each** $x \in$ four roots **do**
- 6: $y \leftarrow -(f_1 x^2 + f_4 x + f_6)/(f_2 x + f_5)$
- 7: apply Newton's method on $f(x, y)$ and $g(x, y)$
- 8: $\boldsymbol{\mu} \leftarrow [1, x, y]^\top$
- 9: $s \leftarrow \|\mathbf{A}\boldsymbol{\mu}\|/a, \quad \boldsymbol{\lambda} \leftarrow \boldsymbol{\mu}/s$
- 10: $\mathbf{r}_1 \leftarrow \mathbf{A}\boldsymbol{\lambda}/a, \quad \mathbf{r}_2 \leftarrow \mathbf{C}\boldsymbol{\lambda}/c$
- 11: $\mathbf{R}' \leftarrow [\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_1 \times \mathbf{r}_2], \quad \mathbf{t}' \leftarrow \lambda_1 \mathbf{m}_1$
- 12: $\mathbf{R} \leftarrow \mathbf{R}' \mathbf{N}^\top, \quad \mathbf{t} \leftarrow \mathbf{t} - \mathbf{R} \mathbf{X}_1$
- 13: **end**

5.3.3 世界座標系における外部パラメータの復元

射影的奥行き之比である x と y が得られると、外部パラメータの復元は容易である。まず、回転行列の各列は単位ベクトルであるという制約 ($\|\mathbf{r}_1\| = \|\mathbf{A}\boldsymbol{\lambda}\|/a = 1$) に基づくと、射影的奥行きは以下のようにスケーリングすればよい。

$$\boldsymbol{\lambda} = \frac{1}{s} \boldsymbol{\mu}, \quad s = \frac{1}{a} \|\mathbf{A}\boldsymbol{\mu}\|. \quad (5.39)$$

そして、 $\boldsymbol{\lambda}$ を式 (5.28) と式 (5.29) に代入すると、回転行列 \mathbf{R}' の第1列 \mathbf{r}_1 と第2列 \mathbf{r}_2 が得られる。第3列は第1列と第2列のクロス積として計算できる。並進ベクトル \mathbf{t}' は式 (5.25) を用いればよい。以上より、中間座標系における回転行列 \mathbf{R}' と並進ベクトル \mathbf{t}' は次式で表される。

$$\begin{aligned} \mathbf{R}' &= [\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_1 \times \mathbf{r}_2], \\ \mathbf{t}' &= \lambda_1 \mathbf{m}_1. \end{aligned} \quad (5.40)$$

最後に、 \mathbf{R}' と \mathbf{t}' を式 (5.24) に代入すると、世界座標系におけるカメラの位置姿勢が以下のように復元できる。

$$\begin{aligned} \mathbf{R} &= \mathbf{R}' \mathbf{N}^\top, \\ \mathbf{t} &= \mathbf{t}' - \mathbf{R} \mathbf{X}_1. \end{aligned} \quad (5.41)$$

提案手法全体の一連の流れを Algorithm 1 に記す。

5.3.4 Banno の手法との違い

5.2.2 節で紹介したように, Banno [7] の手法も同様に回転行列の直交制約を用いて射影的奥行きについての 4 次式に帰着させる. 導出こそ異なるものの, 実は提案手法と Banno [7] の手法は数学的には等価である.

最も大きな違いは, どの射影的奥行き λ_i を 1 とするか, である. 提案手法では λ_1 ^{*4} を, Banno は λ_3 を選択している. これはごく僅かな違いに思えるが, 実際にはそうではない. A の第 3 列と B の第 2 列がゼロベクトルであるため, $\lambda_1 = 1$ と選択することで $f(x, y)$ と $g(x, y)$ の係数を削減できる. 例えば, 付録 5.6 の式 (5.43) に示すように, もし $\lambda_3 = 1$ とした場合は y^2 の係数はゼロにはならない. さらに, 同じく付録 5.6 の (5.44) に示すように, 式 (5.33) の導出において, 式 (5.32) の結果を用いて係数の削減を行っている. 以上のことから, 提案手法は Banno の手法の最適化とも解釈できる. これらの一見小さな違いが性能に及ぼす影響については, 次節のシミュレーション実験にて実証する.

5.4 シミュレーション実験

本節ではシミュレーションデータによる従来手法との比較評価実験について報告する. 評価項目は, 数値的安定性, ノイズに対する頑健性, 処理速度の 3 点である.

提案手法は MATLAB にて実装した^{*5}. 従来手法については, Kneip ら [37] については公式の MATLAB 実装^{*6}を, その他の手法については C++ 実装を忠実に MATLAB に移植した. Gao ら [20] と Ke [35] の手法は OpenCV のコード^{*7}を, Banno [7]^{*8}と Persson [59]^{*9}は著者による Github のコードを参照した. Persson 以外の手法はいずれも 4 次式を解くために独自の実装がされていたが, 公正な比較のために Kneip による Ferrari の公式の実装に統一した.

シミュレーションデータは以下のように生成した. まず, カメラ配置は Kneip ら [37] の実験設定に従い, $\mathbf{R} = \text{diag}([1, -1, -1])$, $\mathbf{t} = [0, 0, 6]^T$, 焦点距離 $f = 800$, 光学中心 $(c_u, c_v) = (320, 240)$ とした. そして, 回転行列の誤差として真値と推定値の四元数の L2 ノルム誤差 $\|\mathbf{q}_{\text{gt}} - \mathbf{q}_{\text{est}}\|$ を, 並進ベクトルの誤差として真値と推定値の相対誤差 $\|\mathbf{t}_{\text{gt}} - \mathbf{t}_{\text{est}}\| / \|\mathbf{t}_{\text{gt}}\|$ をそれぞれ計測した.

^{*4}提案手法においては λ_2 を選択してもよい.

^{*5}<https://github.com/g9nkn/p3p-problem>

^{*6}https://dl.dropboxusercontent.com/u/23966023/home_page_files/p3p_code_final.zip

^{*7} solveP3P function, https://docs.opencv.org/4.1.0/d9/d0c/group_calib3d.html

^{*8}<https://github.com/atsuhikobanno/p3p>

^{*9}<https://github.com/midjji/lambda-twist-p3p>

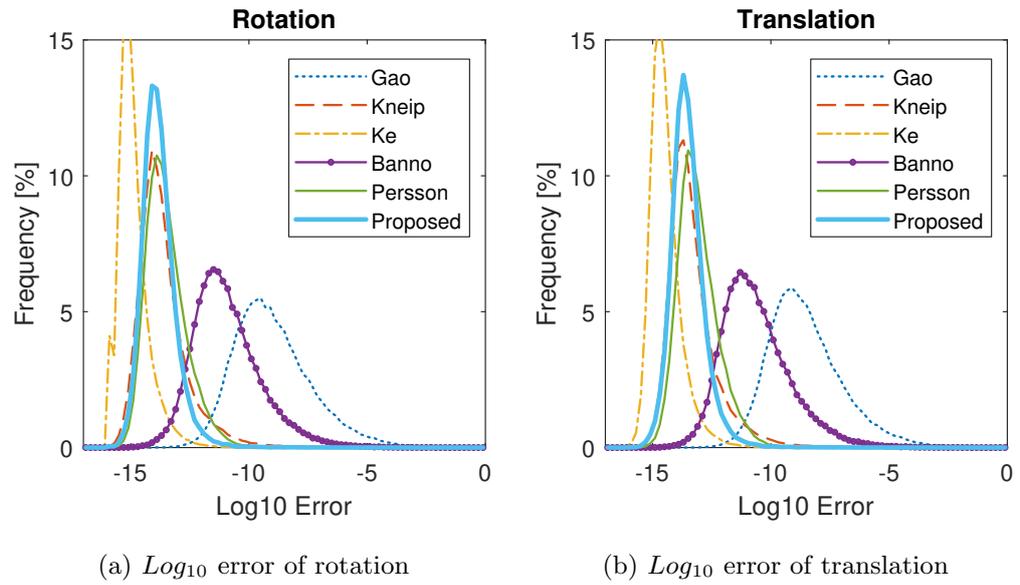


図 5.3: 一般的な点配置における数値演算誤差のヒストグラム

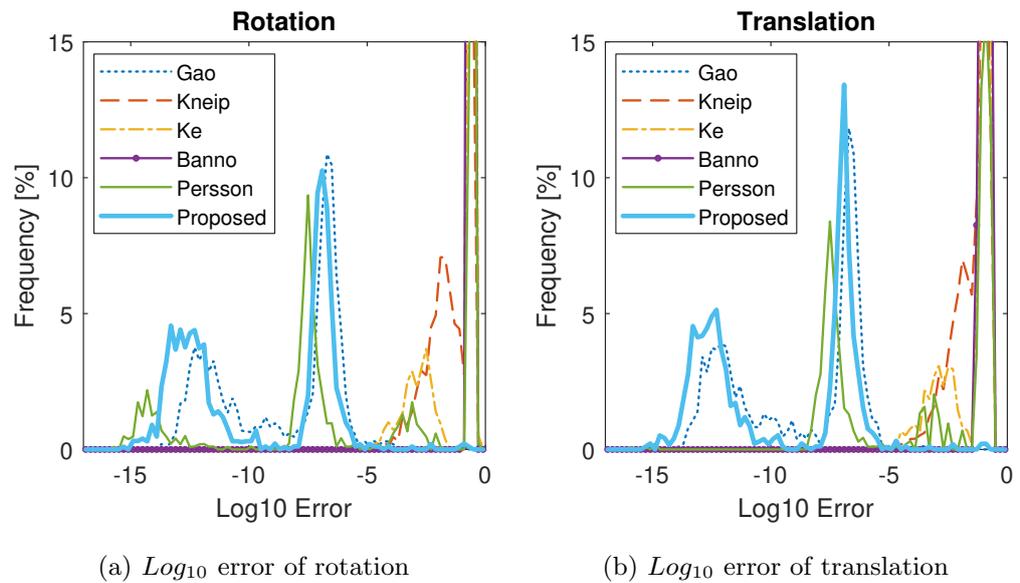


図 5.4: 退化条件における数値演算誤差のヒストグラム

5.4.1 数値的安定性

本実験では、2種類の3次元点の分布について実験を行い、数値的安定性を評価した。1つ目の分布は、 x - y - z が $[-2, 2] \times [-2, 2] \times [-2, 2]$ の範囲内に一様分布に従い 1000 点を生成し、試行のたびにその中からランダムに3点を選択した。2つ目の分布は、Kneipらの手法が数値的に縮退する条件を設定した。具体的には、中間座標系において \mathbf{X}'_2 と \mathbf{X}'_3

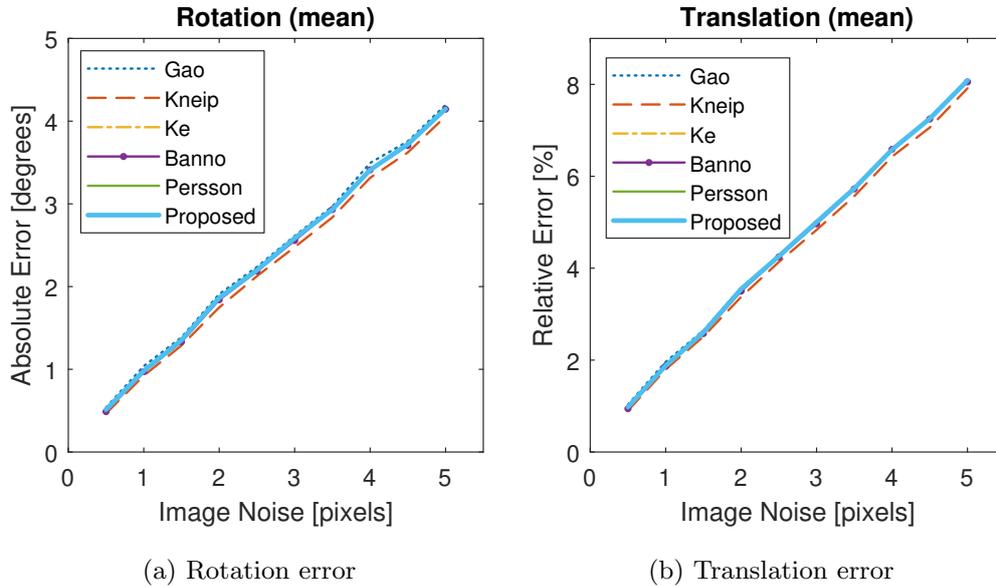


図 5.5: 画像ノイズの変化に対する平均推定誤差

の x 座標が同じ値, つまり, $a = b$ になるランダムな 3 点である. このとき, 文献 [37] の式 (10) において分母がゼロになるとなるため, 解を求めることができなくなる. どちらの分布においても射影した画像座標にはノイズを加えず, 観測誤差がないことを仮定した.

独立に 10^5 回試行した数値演算誤差のヒストグラムを図 5.3 と図 5.4 に示す. それぞれ第 1 と第 2 の 3 次元点分布に相当する.

第 1 の点分布においては, Ke らの手法がピーク値 10^{-15} と最も安定性が高い結果となり, 続いて提案手法, Kneip, Persson がピーク値 10^{-14} である. Ke らの手法は最も高精度であるが, ピーク値の差はごく僅かであるため実用上は同性能と言える. 一方で, 第 2 の点分布においては提案手法と Gao らの手法以外は真値に近い推定値を算出できていない. 第 2 の点分布は Kneip らの手法を主眼に設定したが, 図 5.4 の結果は Ke ら, Banno, Persson らの手法いずれも同様の数値的な退化条件がアルゴリズムに内在することを示唆している.

第 2 の点分布においては, カメラの方向が 3 点 $\mathbf{X}_{i \in \{1,2,3\}}$ の作る平面と正対する. 言い換えると, カメラの z 軸と平面の法線ベクトルが互いに平行となる. 例えば正方 AR マーカを正面から撮影するなど, 実用上このような状況は決して珍しいことではなく, カメラ位置姿勢の推定が不安定になる条件の 1 つとして知られている. 以上のことから, P3P 問題においては 3 次元点群とカメラの位置関係がもたらす数値的安定性についてはいまだ議論の余地があることがわかる.

表 5.1: 独立に 10^5 回試行した平均実行時間

Method	Time [μsec]
Gao [20]	126.6
Kneip [37]	88.8
Ke [35]	72.1
Banno [7]	79.2
Persson [59]	51.6
Proposed	63.3

表 5.2: 提案手法の実行時間プロファイル. *Step*: 各処理ステップ. *Line No.*: Algorithm 1 の行番号. *Time*: 実行時間 (マイクロ秒) *Rate*: 処理全体に対する割合

Step	Line No.	Time	Rate
Preprocess	1–3	14.5	23%
Root finding	4, 6	9.5	15%
Root polishing	7	8.9	14%
Recover pose	8–12	30.4	48%
Total		63.3	100%

5.4.2 ノイズ頑健性

本実験では、点分布は一般的なランダム配置（前節の第1の点分布）とし、画像座標に平均ゼロの正規分布ノイズを加えた。そして、ノイズの標準偏差を0.5ピクセルから5ピクセルへと変化させながら回転行列と並進ベクトルの推定誤差を計測した。

各ノイズレベルについて独立に 10^5 回試行した結果を図 5.5 に示す。Kneip らの手法がわずかながら他の手法よりも良いように見えるが、その差は実用上無視できるほど小さい。

この実験結果は、第2の点分布のような数値的に縮退しない条件であれば、いずれの手法を選んでも実用上はほぼ同一の性能であることを示している。

5.4.3 処理速度

本実験では Core i9-7920X を搭載する PC において独立に 10^5 回試行して処理速度を測定した。表 5.1 に各手法の平均実行時間を示す。最も高速なのは、4次式ではなく3次式を解く Persson らの手法である。4次式を解く手法の中では提案手法が最も速く、Banno より 20%、Ke より 12%、Kneip より 28%、Gao よりも 50% 高速であった。

提案手法の実行プロファイルを表 5.2 に示す。最も実行時間を要するのは4次式を解いた後の最終的なカメラ位置姿勢の復元であり、全体の 48% を占めている。Root polishing (ニュートン法) による解の改善は比較的割合が小さいため、それ自体はトレードオフの問題にはなっていない。

5.4.4 考察

数値的安定性 (5.4.1 節) とノイズ頑健性 (5.4.2 節) より, $a = b$ の退化条件を除けば提案手法も含めて直接解法はいずれもほぼ同性能とみなせることが分かった. 一方で, 実行速度 (5.4.3 節) と退化条件を考慮に入れると, 提案手法が数値的安定性と処理速度のバランスが最も優れていると結論づけられる.

5.5 第5章のまとめ

本章では, まず, P3P 問題に対して非直接解法と直接解法の2種類が従来提案されてきたことを紹介し, 代表的な手法について導出の特徴を紹介した. 次に, 直接解法の1つとして回転行列を射影的奥行き線形結合とみなす新たな手法を提案した. 幾何的に複雑な解析を要する従来手法とは異なり, 提案手法は代数的な制約条件に基づくため, その導出は非常にシンプルである. また, 提案手法と Banno の手法との数学的な類似性と相違点について考察した. 最後に, シミュレーション実験を行い, 提案手法が4次式を用いる手法の中では数値的安定性と処理速度のバランスが最も取れた手法であること実証した. また, 従来手法にはアルゴリズムが退化して解が求まらない3次元点群とカメラ配置があることを実験的に示し, P3P 問題にはまだ議論の余地があることを示した.

本章で議論しなかった点は, 内部パラメータと3次元座標に対するノイズ頑健性である. P3P 問題では内部パラメータと3次元座標は高精度に計測済みと仮定しているが, 実際には誤差が含まれるはずである. 特に3次元座標の中でも x , y に比べて z 方向の計測誤差は大きくなりやすく, その誤差分布は計測方法 (例えばステレオ画像による3角測量かレーザーレンジファインダか) により異なる. この点については従来の P3P 問題においても十分に議論されていないため, 今後も研究の余地が残されている.

本章に関連する公表論文は以下の通りである.

Gaku Nakano, A Simple Direct Solution to the Perspective-three-Point Problem, the 30th British Machine Vision Conference (BMVC2019), 2019.

本章にて提案した手法のコードは以下のウェブサイトにて公開している.

<https://github.com/g9nkn/p3p-problem>

5.6 付録：提案手法の導出詳細

式 (5.32) : $f(x, y)$ の導出

$$\begin{aligned}
 \mathbf{r}_1^\top \mathbf{r}_2 &= \frac{1}{ac} \boldsymbol{\mu}^\top \mathbf{A}^\top \mathbf{C} \boldsymbol{\mu} \\
 &= \frac{1}{ac} \boldsymbol{\mu}^\top \mathbf{A}^\top (\mathbf{B} - p\mathbf{A}) \boldsymbol{\mu} \\
 &\propto \boldsymbol{\lambda}^\top (p\mathbf{A}^\top \mathbf{A} - \mathbf{A}^\top \mathbf{B}) \boldsymbol{\lambda}
 \end{aligned} \tag{5.42}$$

$$\begin{aligned}
 f(x, y) &= \boldsymbol{\lambda}^\top (p\mathbf{A}^\top \mathbf{A} - \mathbf{A}^\top \mathbf{B}) \boldsymbol{\lambda} \\
 &= [1, x, y] \left(p \begin{bmatrix} 1 & -\mathbf{m}_1^\top \mathbf{m}_2 & 0 \\ -\mathbf{m}_1^\top \mathbf{m}_2 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 1 & 0 & -\mathbf{m}_1^\top \mathbf{m}_3 \\ -\mathbf{m}_1^\top \mathbf{m}_2 & 0 & \mathbf{m}_2^\top \mathbf{m}_3 \\ 0 & 0 & 0 \end{bmatrix} \right) \begin{bmatrix} 1 \\ x \\ y \end{bmatrix} \\
 &= [1, x, y] \begin{bmatrix} p-1 & -p(\mathbf{m}_1^\top \mathbf{m}_2) & \mathbf{m}_1^\top \mathbf{m}_3 \\ (1-p)(\mathbf{m}_1^\top \mathbf{m}_2) & p & -\mathbf{m}_2^\top \mathbf{m}_3 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ x \\ y \end{bmatrix} \\
 &= px^2 - (\mathbf{m}_2^\top \mathbf{m}_3)xy + (1-2p)(\mathbf{m}_1^\top \mathbf{m}_2)x + (\mathbf{m}_1^\top \mathbf{m}_3)y + p - 1
 \end{aligned} \tag{5.43}$$

式 (5.33) : $g(x, y)$ の導出

式 (5.42) より, $\boldsymbol{\mu}^\top \mathbf{A}^\top \mathbf{C} \boldsymbol{\mu} = 0 \Leftrightarrow \boldsymbol{\mu}^\top \mathbf{B}^\top \mathbf{A} \boldsymbol{\mu} = p \boldsymbol{\mu}^\top \mathbf{A}^\top \mathbf{A} \boldsymbol{\mu}$ であることを利用する.

$$\begin{aligned}
 \mathbf{r}_1^\top \mathbf{r}_1 - \mathbf{r}_2^\top \mathbf{r}_2 &= \frac{1}{a^2} \boldsymbol{\mu}^\top \mathbf{A}^\top \mathbf{A} \boldsymbol{\mu} - \frac{1}{c^2} \boldsymbol{\mu}^\top \mathbf{C}^\top \mathbf{C} \boldsymbol{\mu} \\
 &= \frac{1}{a^2} \boldsymbol{\mu}^\top \mathbf{A}^\top \mathbf{A} \boldsymbol{\mu} - \frac{1}{c^2} \boldsymbol{\mu}^\top (\mathbf{B} - p\mathbf{A})^\top (\mathbf{B} - p\mathbf{A}) \boldsymbol{\mu} \\
 &= \frac{1}{a^2} \boldsymbol{\mu}^\top \mathbf{A}^\top \mathbf{A} \boldsymbol{\mu} - \frac{1}{c^2} \boldsymbol{\mu}^\top \mathbf{B}^\top \mathbf{B} \boldsymbol{\mu} + \frac{p}{c^2} \boldsymbol{\mu}^\top \mathbf{B}^\top \mathbf{A} \boldsymbol{\mu} + \frac{p}{c^2} \boldsymbol{\mu}^\top \mathbf{A}^\top \mathbf{C} \boldsymbol{\mu} \\
 &= \frac{1}{a^2} \boldsymbol{\mu}^\top \mathbf{A}^\top \mathbf{A} \boldsymbol{\mu} - \frac{1}{c^2} \boldsymbol{\mu}^\top \mathbf{B}^\top \mathbf{B} \boldsymbol{\mu} + \frac{p^2}{c^2} \boldsymbol{\mu}^\top \mathbf{A}^\top \mathbf{A} \boldsymbol{\mu} \\
 &= \frac{1}{c^2} \boldsymbol{\mu}^\top \left(\frac{b^2 + c^2}{a^2} \mathbf{A}^\top \mathbf{A} - \mathbf{B}^\top \mathbf{B} \right) \boldsymbol{\mu} \\
 &\propto \boldsymbol{\lambda}^\top (q\mathbf{A}^\top \mathbf{A} - \mathbf{B}^\top \mathbf{B}) \boldsymbol{\lambda}
 \end{aligned} \tag{5.44}$$

$$\begin{aligned}
g(x, y) &= \boldsymbol{\lambda}^\top (q\mathbf{A}^\top \mathbf{A} - \mathbf{B}^\top \mathbf{B}) \boldsymbol{\lambda} \\
&= [1, x, y] \left(q \begin{bmatrix} 1 & -\mathbf{m}_1^\top \mathbf{m}_2 & 0 \\ -\mathbf{m}_1^\top \mathbf{m}_2 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 1 & 0 & -\mathbf{m}_1^\top \mathbf{m}_3 \\ 0 & 0 & 0 \\ -\mathbf{m}_1^\top \mathbf{m}_3 & 0 & 1 \end{bmatrix} \right) \begin{bmatrix} 1 \\ x \\ y \end{bmatrix} \\
&= [1, x, y] \begin{bmatrix} q-1 & -q(\mathbf{m}_1^\top \mathbf{m}_2) & \mathbf{m}_1^\top \mathbf{m}_3 \\ -q(\mathbf{m}_1^\top \mathbf{m}_2) & q & 0 \\ \mathbf{m}_1^\top \mathbf{m}_3 & 0 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ x \\ y \end{bmatrix} \\
&= qx^2 - y^2 - 2q(\mathbf{m}_1^\top \mathbf{m}_2)x + 2(\mathbf{m}_1^\top \mathbf{m}_3)y + q - 1
\end{aligned} \tag{5.45}$$

第6章

PnP 問題への提案手法

本章では PnP 問題について議論する。P3P 問題は $n \geq 3$ とした P3P 問題の一般的な拡張である。まず、PnP 問題の定式化を行い、次に、代表的な従来手法を簡単に説明する。従来手法は、線形最小二乗法に基づく方法、交互最適化に基づく方法、凸最適化に基づく方法、グレブナー基底を用いる方法を紹介する。そして、提案手法の詳細な導出を行い、シミュレーション実験にて従来手法との性能を比較する。

6.1 PnP 問題の定式化

本節では 4.7 節の式 (4.17) にて説明した代数的誤差をもとに PnP 問題の定式化を行う。PnP 問題は P3P 問題と同様にカメラは事前に校正済みで内部パラメータは既知とする。そのため、画像座標 \mathbf{m} は内部パラメータ行列 \mathbf{K} を用いて $\mathbf{m} \leftarrow \mathbf{K}^{-1}\mathbf{m}$ と正規化画像座標系へと変換されているものとする。ただし、P3P 問題とは異なりノルム正規化については必須ではない。

代数的誤差を用いると、回転行列 \mathbf{R} と並進ベクトル \mathbf{t} を未知数とする PnP 問題は以下の最適化問題として表される。

$$\begin{aligned} \min_{\mathbf{R}, \mathbf{t}} \quad & \sum_{i=1}^n \left\| [\mathbf{m}_i]_{\times} (\mathbf{R}\mathbf{X}_i + \mathbf{t}) \right\|^2 \\ \text{s.t.} \quad & \mathbf{R}^T \mathbf{R} = \mathbf{I}, \quad \det(\mathbf{R}) = 1 \end{aligned} \quad (6.1)$$

ここで、ベクトルのクロス積を行列演算により表記している。

$$\mathbf{a} \times \mathbf{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \mathbf{b} = [\mathbf{a}]_{\times} \mathbf{b} \quad (6.2)$$

式 (6.2) が示すように、 $\text{rank}([\mathbf{m}_i]_{\times}) = 2$ であるから、1つの対応点組 $\{\mathbf{m} \leftrightarrow \mathbf{X}\}$ から得られる拘束条件は2つである。PnP 問題の未知数は計6個であるから、式 (6.1) を解くのに必要な対応点数は $n \geq 3$ である。

次に、式 (6.1) が回転行列のみを変数とする等価な最適化問題として表せることを示

す。まず，行列表記を用いると目的関数は次式のように書き換えられる。

$$\begin{aligned} \sum_{i=1}^n \|\mathbf{m}_i \times (\mathbf{R}\mathbf{X}_i + \mathbf{t})\|^2 &= \sum_{i=1}^n \|\mathbf{X}_i \mathbf{Y}_i \mathbf{r} + \mathbf{X}_i \mathbf{t}\|^2 \\ &= \|\mathbf{A}\mathbf{r} + \mathbf{B}\mathbf{t}\|^2 \end{aligned} \quad (6.3)$$

ここで， \mathbf{r} は \mathbf{R} の 9 次元ベクトル表記であり，

$$\begin{aligned} \mathbf{X}_i &= \begin{bmatrix} 0 & -1 & v_i \\ 1 & 0 & -u_i \end{bmatrix}, \quad \mathbf{Y}_i = \begin{bmatrix} \mathbf{X}_i^\top & & \\ & \mathbf{X}_i^\top & \\ & & \mathbf{X}_i^\top \end{bmatrix} \\ \mathbf{A} &= \begin{bmatrix} \mathbf{X}_1 \mathbf{Y}_1 \\ \vdots \\ \mathbf{X}_n \mathbf{Y}_n \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_n \end{bmatrix} \end{aligned} \quad (6.4)$$

である。並進ベクトル \mathbf{t} に関しては何ら制約条件がないため，式 (6.3) は \mathbf{t} についての線形最小二乗問題とみなせる。すなわち， \mathbf{t} の 1 次最適性条件は，式 (6.3) を微分してゼロとおけばよいから，

$$\mathbf{B}^\top (\mathbf{A}\mathbf{r} + \mathbf{B}\mathbf{t}) = \mathbf{0} \quad (6.5)$$

と表せる。ゆえに， \mathbf{t} の最適解は

$$\mathbf{t} = -(\mathbf{B}^\top \mathbf{B})^{-1} \mathbf{B}^\top \mathbf{A}\mathbf{r} \quad (6.6)$$

である。次に，式 (6.6) を式 (6.3) に代入すると，目的関数は次式で表される。

$$\begin{aligned} \|\mathbf{A}\mathbf{r} + \mathbf{B}\mathbf{t}\|^2 &= \|(\mathbf{I} - \mathbf{B}(\mathbf{B}^\top \mathbf{B})^{-1} \mathbf{B}^\top) \mathbf{A}\mathbf{r}\|^2 \\ &= \mathbf{r}^\top \mathbf{M}\mathbf{r} \end{aligned} \quad (6.7)$$

ただし，

$$\mathbf{M} = \mathbf{A}^\top \mathbf{A} - \mathbf{A}^\top \mathbf{B}(\mathbf{B}^\top \mathbf{B})^{-1} \mathbf{B}^\top \mathbf{A} \quad (6.8)$$

とおいた。以上より，式 (6.1) は以下の等価な最適化問題として表せる。

$$\begin{aligned} \min_{\mathbf{R}, \mathbf{t}} \quad & \mathbf{r}^\top \mathbf{M}\mathbf{r} \\ \text{s.t.} \quad & \mathbf{R}^\top \mathbf{R} = \mathbf{I}, \quad \det(\mathbf{R}) = 1 \end{aligned} \quad (6.9)$$

6.2 従来手法

6.2.1 Direct Linear Transform (DLT) 法

Direct Linear Transform (DLT) 法とは，非線形制約付き最適化問題を解くためのコンピュータビジョンにおいては標準的な手法である [24]。基本的なアプローチは，非線形制約を単なる L2 ノルム制約と置き換えて線形最小二乗法として解き，次に制約を満たすようにパラメータを補正する。実装が簡単なため，再投影誤差を最小化するための初期値を求める方法として広く使われている。

PnP 問題における DLT 法について説明する．前節では回転行列のみの最適化問題として定式化したのが，文献 [24] にならって式 (6.3) に基づいて説明する．式 (6.3) の目的関数を利用して回転行列の制約を L2 ノルム制約へと置き換えると，次式を得る．

$$\begin{aligned} \min_{\mathbf{p}} \quad & \|\mathbf{Cp}\|^2 \\ \text{s.t.} \quad & \|\mathbf{p}\|^2 - 1 = 0 \end{aligned} \quad (6.10)$$

ただし，

$$\mathbf{C} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \end{bmatrix}, \quad \mathbf{p} = \begin{bmatrix} \mathbf{r} \\ \mathbf{t} \end{bmatrix} \quad (6.11)$$

とおいた．未定乗数を λ とすると式 (6.10) のラグランジュ関数は

$$L = \mathbf{p}^T \mathbf{C}^T \mathbf{C} \mathbf{p} - \lambda (\mathbf{p}^T \mathbf{p} - 1) \quad (6.12)$$

と書ける．式 (6.12) の解は，ラグランジュ関数 L を \mathbf{p} にて偏微分した極値であるから，

$$\frac{\partial L}{\partial \mathbf{p}} = \mathbf{C}^T \mathbf{C} \mathbf{p} - \lambda \mathbf{p} = \mathbf{0} \rightarrow \mathbf{C}^T \mathbf{C} \mathbf{p} = \lambda \mathbf{p} \quad (6.13)$$

を解けばよい．上式は $\mathbf{C}^T \mathbf{C}$ についての固有値問題にほかならない．式 (6.10) の目的関数値は

$$\mathbf{p}^T \mathbf{C}^T \mathbf{C} \mathbf{p} = \lambda \mathbf{p}^T \mathbf{p} = \lambda \quad (6.14)$$

であるから，式 (6.10) の最適解は， $\mathbf{C}^T \mathbf{C}$ の最小固有値 λ_{\min} に対応する固有ベクトル \mathbf{p} である．固有ベクトル \mathbf{p} の上 9 成分から \mathbf{r} ，下 3 成分から \mathbf{t} が得られる．そして， $\|\mathbf{r}\| = \sqrt{3}$ となるように \mathbf{r} と \mathbf{t} をスケールリングし，得られた \mathbf{r} を 3×3 に変形すれば行列 \mathbf{R} となる．しかし，この段階では行列 \mathbf{R} は直交制約を満たしていないことに注意する．そこで，式 (4.14) に示した方法により，制約を満たすように補正する．

上述したように，DLT 法は線形最小二乗法と解の補正からなるため，制約付き非線形最適化である原問題よりも実装が簡単という利点がある．しかしながら，元の制約を無視した線形化による 2 つの問題点がある．

1 つは，3 次元点が平面か非平面かで場合分けが必要なことである．本節ではこれまで暗黙的に 3 次元点が非平面であることを仮定していた．もし 3 次元点がすべて $z = 0$ の平面上の点とすると，式 (6.4) から行列 \mathbf{A} の第 3 列，第 6 列，第 9 列がすべてゼロベクトルになることが分かる．つまり， $\text{rank}(\mathbf{C}^T \mathbf{C}) = 6$ となるため， $\mathbf{p} = \text{span}(\text{null}(\mathbf{C}^T \mathbf{C}))$ であるから 1 つの固有ベクトルに定めることができない．平面の場合には， \mathbf{A} からゼロベクトルを除き， \mathbf{r} の第 3, 6, 9 成分を除いた 6 次元ベクトルとして計算する必要がある．平面性は式 (4.15) に示した方法で判定できるが，准平面の場合どちらの手法を選択するかが重要となる．

もう 1 つの欠点は，式 (4.14) による回転行列の補正は原問題の目的関数をなんら考慮していないことである．つまり，補正により目的関数が増幅されてしまい，再投影誤差最小化の初期値として適当でないという可能性もある．

6.2.2 交互最適化法

Lu ら [50] によって提案された交互最適化法は以下の通りである。

1. 初期 R_0 を選ぶ. (k 回目の推定値を R_k , t_k とする)
2. 式 (6.6) に R_k を代入し, t_k を計算する.
3. 式 (6.3) に t_k を代入し, 最小二乗法により r_{k+1} を計算する.
4. 式 (4.14) に基づき, r_{k+1} を変形した R_{k+1} を回転行列へ補正する.
5. ステップ 2–4 を推定値の差分が閾値以下になるまで反復する.

上記の反復法はどのような初期値から始めても何らかの局所解へ収束することが証明されている. 初期値として Lu らは弱透視投影に基づいて R_0 を選ぶ方法を推奨しているが, その後様々な論文で不適切な局所解へ陥ることが実験的に示された. 後に, Schweighofer and Pinz [68] は, 平面に限定することで大域的最適解に収束するように改良した.

6.2.3 EPnP 法

Moreno-Noguer ら [52] は, 点数 n の増加に対して計算量が線形時間 $\mathcal{O}(n)$ である EnP 法を提案した. EPnP 方は, 3次元座標 X_i を4つの制御点 $c_j, j \in 1, \dots, 4$ の線形結合で表現する新たな座標系を定義し, PnP 問題を制御点 c_j の推定問題として定式化した. 座標系の定義は異なるものの, 実質的には式 (6.10) で表される DLT 法に類似した最適化問題を解く. DLT 法にない特長は, $\text{rank}(C^T C)$ に応じて4種類の解法を提案していることである. 4種類の解法をすべて適用して最も再投影誤差が小さい解を選出する. さらに, カメラ座標系における制御点間の距離は世界座標系の原点周りの単位ベクトル間の距離に等しいことを利用して, 制御点の非線形最適化を行い, 解の精度を向上させる. 計算量が $\mathcal{O}(n)$ であるのは, $C^T C$ が n に対して線形増加であるためである. 高速ではあるが, 平面と非平面を区別しなくてはならない.

6.2.4 RPnP 法

Li ら [48] は P3P 問題の非直接解法を $n \geq 3$ へ一般化した手法を提案した. まず, 全 n 点の3次元点を, ある2点を共有する3点ずつの組に分割し, その2点を中心とする中間座標系において P3P 問題の式 (5.7) を $n-2$ 本立てる. そして, $n-2$ 本の式を最小二乗法により解いて2点への射影的奥行きを算出し, 中間座標系における回転軸を決定する. 最後に, DLT 法に似たアプローチで回転角度と並進ベクトルを算出する.

6.2.5 凸最適化に基づく手法

凸最適化に基づく手法 [27, 69] では四元数を用いて式 (6.9) を半正定値計画へと変換する。理論的には大域的最適解への収束が保証されている。導出は複雑なため本節では省略する。

6.2.6 グレブナー基底を用いた手法

グレブナー基底 (Gröbner basis) とは、多変数連立多項式を解析するために計算機代数学で提唱された理論および方法である [14, 96]。多項式 (polynomial) とは、 x^2 や $2xy$ のような単項式 (monomial) の積和で表される式のことである。コンピュータビジョン分野において 2000 年代後半から 2 視点幾何問題を解くために導入され [58, 73, 74]、その成功を受けて PnP 問題にも利用されるに至った。グレブナー基底の理論は複雑であるため、本論文では簡単な具体例の紹介に留める。

ある n 変数の多項式集合 $F = f_1, \dots, f_m$ があったとき、 F と「等価」かつ「性質がよい」一意な多項式集合 $G = g_1, \dots, g_r$ のことをグレブナー基底と呼ぶ。ここで、「等価」とは F と G は同じ解空間を持つ、つまり、 $f_1 = 0, \dots, f_m = 0$ の零点と $g_1 = 0, \dots, g_r = 0$ の零点は等しいことを表す。また、「性質がよい」とは、ある f_i について任意の多項式 h_i を用いて $f_i = h_i g_1 + \dots + h_r g_r$ と表すことができるということである。 G は一意な集合であるから、 F の零点 (連立多項式の解) を計算するために簡易な表現である $G = 0$ を解けばよい。

グレブナー基底の計算には高い数値演算精度が要求され、計算機代数学では有理数体を用いてグレブナー基底を計算する。しかしながら、コンピュータビジョンにおいてはリアルタイム性を要求されるアプリケーションも多いため、画像が入力されるたびにグレブナー基底を計算することは現実的ではない。そこで、事前に有理数体を用いたオフライン処理でグレブナー基底を得るための係数行列 (elimination template) を構築し、オンライン処理では係数行列に実際の値を代入し浮動小数点で Gauss-Jordan 消去によりグレブナー基底を計算する手法 [42, 38]^{*1*}^{*2}が考案された。問題の定式化さえ行えばグレブナー基底が自動生成されるため、手計算では困難な多項式問題も容易に扱うことができる。一方で、同じ問題であっても elimination template の大きさはパラメータ表現に大きく依存する。オンライン処理は浮動小数点演算であるため、elimination template のサイズによっては演算精度が不足する。それゆえ、自動化ツールを用いる鍵は効率的なパラメータ表現にある。

^{*1}<https://github.com/PavelTrutman/Automatic-Generator>

^{*2}http://people.inf.ethz.ch/vlarsson/misc/autogen_v0.5.zip

DLS 法

Hesch ら [26] と中野ら [94, 95] は, グレブナー基底を用いた最初の手法 (Direct Least Square Method, DLS 法) を提案した. 式 (6.9) の回転行列を Cayley 変換で表し, 無制約最適化とみなしてその 1 次最適性条件をグレブナー基底で解く. Cayley 変換は式 4.4.4 に示すとおり, 最初の成分を 1 とおいた四元数 ($\mathbf{q} = [1, b, c, d]^T$) に相当する. 1 次最適性条件である 3 変数連立多項式は次式で表せる.

$$\left[\frac{\partial \mathbf{r}}{\partial b}, \frac{\partial \mathbf{r}}{\partial c}, \frac{\partial \mathbf{r}}{\partial d} \right]^T \mathbf{M} \mathbf{r} = \mathbf{0}_{3 \times 1} \quad (6.15)$$

式 (6.15) を元に, Hesch らは手計算で 120×120 の, 中野らは自動化ツール [38] を用いて 89×117 の elimination template を生成した. Gauss-Jordan 消去を適用後, 両手法ともに 27×27 の固有値分解により最大 27 個の実数解を算出する. DLS 法の問題点は, 本来 Cayley 変換に必要な回転行列のスケーリング $1/\|\mathbf{q}\|^2$ を無視していることである.

OPnP 法

Zheng ら [91] は, 非単位四元数 $\mathbf{q} = [a, b, c, d]^T$ (ただし $\|\mathbf{q}\| \neq 1$) を用いることで, 四元数の L2 ノルム制約を使わずに PnP 問題を無制約最適化問題として定式化した. 具体的には, 四元数の L2 ノルム $\|\mathbf{q}\|$ を射影的奥行き $\bar{\lambda} = 1/n \sum \lambda_i$ の逆数とおく. つまり, $\|\mathbf{q}\| = 1/\bar{\lambda}$ として式 (6.9) のように \mathbf{r} のみで表される無制約最適化問題を導出した.

$$\min_{\mathbf{q}} f(\mathbf{q}) \quad (6.16)$$

上式において目的関数 $f(\mathbf{q})$ は式 (6.9) と若干異なるものほぼ同じである. 1 次最適性条件は

$$\frac{\partial f}{\partial \mathbf{q}} = \mathbf{0}_{4 \times 1} \quad (6.17)$$

である. そして, Ask ら [6] の手法を用いて四元数の符号不定性 $\pm \mathbf{q}$ を除去した結果, 348×376 の elimination template と 40×40 の固有値分解に帰着させた. 実数解の数は最大 40 個である.

UPnP 法

Kneip ら [36] は, 正規化四元数 $\mathbf{q} = [a, b, c, d]^T$ (ただし $\|\mathbf{q}\| = 1$) を用いた解法を提案した. Kneip らは四元数の L2 ノルムを 2 乗した制約条件を導入して PnP 問題を以下のように定式化した.

$$\begin{aligned} \min_{\mathbf{q}} \quad & \mathbf{r}^T \mathbf{M} \mathbf{r} \\ \text{s.t.} \quad & (\mathbf{q}^T \mathbf{q} - 1)^2 = 0 \end{aligned} \quad (6.18)$$

上式の 1 次最適性条件は

$$\begin{aligned} \begin{pmatrix} \frac{\partial \mathbf{r}}{\partial \mathbf{q}} \end{pmatrix}^T \mathbf{M}\mathbf{q} &= \mathbf{0}_{4 \times 1} \\ (\mathbf{q}^T \mathbf{q} - 1)\mathbf{q} &= \mathbf{0}_{4 \times 1} \end{aligned} \quad (6.19)$$

となる。未定乗数が現れないのは、 $(\mathbf{q}^T \mathbf{q} - 1)$ の項が未定乗数にかかり消去されるためである。その結果、elimination template のサイズは 141×149 と OPnP 法よりも小さくなり、 8×8 の固有値分解により最大 8 個の実数解を算出する。

6.3 提案手法

6.3.1 未定乗数を用いない 1 次最適性条件

本節では、未定乗数を用いずに式 (6.9) で表される制約付き最適化問題の新しい 1 次最適性条件を導出する。式 (6.9) のラグランジュ関数は次式で表される。

$$L(\mathbf{R}, \mathbf{S}, \lambda) = \frac{1}{2} \mathbf{r}^T \mathbf{M}\mathbf{r} - \frac{1}{2} \text{trace}(\mathbf{S}(\mathbf{R}^T \mathbf{R} - \mathbf{I})) - \lambda(\det(\mathbf{R}) - 1). \quad (6.20)$$

ここで、 λ は未定乗数、 \mathbf{S} は未定乗数を成分とする 3×3 の対称行列であり、 $1/2$ は便宜的な係数である。1 次最適性条件は以下のように書ける。

$$\frac{\partial L}{\partial \mathbf{R}} = \text{mat}(\mathbf{M}\mathbf{r}) - \mathbf{R}\mathbf{S} - \lambda\mathbf{R} = \mathbf{0} \quad (6.21)$$

$$\frac{\partial L}{\partial \mathbf{S}} = \mathbf{R}^T \mathbf{R} - \mathbf{I} = \mathbf{0} \quad (6.22)$$

$$\frac{\partial L}{\partial \lambda} = \det(\mathbf{R}) - 1 = 0 \quad (6.23)$$

ここで、 $\text{mat}(\)$ は 9 次元ベクトルを 3×3 の正方行列へ変形するオペレータである。式 (6.21) より次式を得る。

$$\text{mat}(\mathbf{M}\mathbf{r}) = \mathbf{R}(\mathbf{S} + \lambda\mathbf{I}). \quad (6.24)$$

上式の右辺と左辺からそれぞれ \mathbf{R}^T かけると、次の 2 式を得る。

$$\mathbf{R}^T \text{mat}(\mathbf{M}\mathbf{r}) = \mathbf{S} + \lambda\mathbf{I} \quad (6.25)$$

$$\text{mat}(\mathbf{M}\mathbf{r}) \mathbf{R}^T = \mathbf{R}(\mathbf{S} + \lambda\mathbf{I}) \mathbf{R}^T \quad (6.26)$$

式 (6.25) の右辺 $\mathbf{S} + \lambda\mathbf{I}$ は対称行列であるから、直交行列を両側からかけた式 (6.26) も同じく対称行列である。ゆえに、式 (6.25) と式 (6.26) の左辺も対称行列でなければならない。対称行列の性質を利用すると、未定乗数が消去された次の 2 式を得る。

$$\mathbf{P} = \mathbf{R}^T \text{mat}(\mathbf{M}\mathbf{r}) - \text{mat}(\mathbf{M}\mathbf{r})^T \mathbf{R} = \mathbf{0}_{3 \times 3} \quad (6.27)$$

$$\mathbf{Q} = \text{mat}(\mathbf{M}\mathbf{r}) \mathbf{R}^T - \mathbf{R} \text{mat}(\mathbf{M}\mathbf{r})^T = \mathbf{0}_{3 \times 3} \quad (6.28)$$

式 (6.27) と式 (6.28) が PnP 問題に対する新しい最適性条件である。ここで、 $P_{i,j}$ と $Q_{i,j}$ を \mathbf{P} と \mathbf{Q} の (i,j) 成分とする。 \mathbf{P} と \mathbf{Q} は対称行列の差であるから、対角成分は明ら

かに $P_{i,i} = Q_{i,i} = 0$ である。一方で、非対角成分は \mathbf{R} に関する 2 次多項式であり、 $P_{i,j} = P_{j,i}$, $Q_{i,j} = Q_{j,i}$ である。ゆえに、以下の 6 個の多項式が得られる。

$$\begin{aligned} P_{1,2} &= 0, & P_{1,3} &= 0, & P_{2,3} &= 0, \\ Q_{1,2} &= 0, & Q_{1,3} &= 0, & Q_{2,3} &= 0. \end{aligned} \quad (6.29)$$

式 (6.29) を回転行列の制約条件である式 (6.22) および式 (6.23) とともに連立させて解けば、PnP 問題の解が得られる。

6.3.2 回転表現ごとの最適性条件

式 (6.29) は一般的な回転行列の制約条件 ($\mathbf{R}^T \mathbf{R} = \mathbf{I}$, $\det(\mathbf{R}) = 1$) から導かれたが、これらの制約を満たすパラメータ表現であれば、PnP 問題を解くために式 (6.29) を用いることができる。Kukelova ら [38] によるグレブナー基底計算ツールを用いて、一般的な回転行列表現と四元数と Cayley 変換の 3 つの表現についてソルバーを生成し、生成されたソルバーの比較を行う。グレブナー基底生成ツールを利用するコードは章末の付録 6.7 に記載した。Kukelova ら [38] のツールに加えて、Larsson ら [42] らのツールを使うためのコードも合わせて記載した。

3 × 3 回転行列

4.4.1 節でも述べたように、通常の 3 × 3 回転行列は 9 個の未知数 r_{11}, \dots, r_{33} と 7 つの制約式 ($\mathbf{R}^T \mathbf{R} = \mathbf{I}$ と $\det(\mathbf{R}) = 1$) で表現される。しかしながら、3 次式の行列式制約は、式 (4.10) に示したような 2 次式で等価に表現できる。低い次数の方がグレブナー基底生成には有利なため、本節では後者を採用する。以上より、用いる制約条件は式 (6.29) に以下の 21 式を加えた計 27 式である。

$$\begin{aligned} \mathbf{R}^T \mathbf{R} - \mathbf{I} &= \mathbf{0}_{3 \times 3}, & \mathbf{R} \mathbf{R}^T - \mathbf{I} &= \mathbf{0}_{3 \times 3}, \\ \mathbf{r}_i - \mathbf{r}_j \times \mathbf{r}_k &= \mathbf{0}, & (i, j, k) &= \begin{cases} (1, 2, 3) \\ (2, 3, 1) \\ (3, 1, 2) \end{cases} \end{aligned} \quad (6.30)$$

ここで、 \mathbf{r} の下添え字 i, j, k は \mathbf{R} の列番号を示す。式 (6.30) を用いて自動生成ツールを利用したところ、elimination template のサイズは 1936×1976 であり、解の個数は 40 個であった。解の個数は OPnP 法と一致しており、パラメータ表現が異なるにも関わらず解空間が互いに同一であることを示唆している。回転行列の表現はそれ自体に退化条件や土の不定性は含んでいない。しかしながら、後述する他の手法と比べて elimination template は最も大きいサイズである。

四元数

単位四元数を $\mathbf{q} = [a, b, c, d]^T$ とすると、回転行列は

$$\mathbf{R} = \begin{bmatrix} a^2 + b^2 - c^2 - d^2 & 2(bc - ad) & 2(bd + ac) \\ 2(bc + ad) & a^2 - b^2 + c^2 - d^2 & 2(cd - ab) \\ 2(bd - ac) & 2(cd + ab) & a^2 - b^2 - c^2 + d^2 \end{bmatrix}. \quad (6.31)$$

と表される。直交性はすでに満たしているため、以下の L2 ノルム制約により $\det(\mathbf{R}) = 1$ となる。

$$\|\mathbf{q}\|^2 - 1 = 0. \quad (6.32)$$

ゆえに、式 (6.29) に上式を加えた計 7 式をグレブナー基底の計算に用いる。自動生成ツールにより、 630×710 の elimination template が生成され、解の個数は 80 個であった。解の個数が回転行列や OPnP 法の 2 倍となっているのは、四元数が符号の不定性を含むためである。つまり、 \mathbf{q} と $-\mathbf{q}$ は同じ回転行列を表すが、Kukelova らの自動生成ツールは符号の不定性を考慮できないため、結果として倍の解空間を探索していることになる。仮に OPnP 法のように Ask ら [6] の手法を利用できれば符号不定性が除外でき、より小さい elimination template が生成できると思われる。

Cayley 変換

Cayley 変換は四元数を a で正規化 ($\mathbf{q} \leftarrow \mathbf{q}/a$) した場合に相当し、回転行列は次式で表される。

$$\mathbf{R} = \frac{1}{s} \begin{bmatrix} 1 + b^2 - c^2 - d^2 & 2(bc - d) & 2(bd + c) \\ 2(bc + d) & 1 - b^2 + c^2 - d^2 & 2(cd - b) \\ 2(bd - c) & 2(cd + b) & 1 - b^2 - c^2 + d^2 \end{bmatrix}, \quad (6.33)$$

ただし、 $s = 1 + b^2 + c^2 + d^2$ である。符号の不定性が除去され、未知数は 4 個から 3 個へと減っている。さらに、直交性と行列式制約はすでに満たしているため、グレブナー基底の計算に用いるのは式 (6.29) の 6 式のみでよい。

ここで、Cayley 変換のスケーリング $1/s$ が解空間に影響を及ぼす懸念がある。しかしながら、 \mathbf{P} と \mathbf{Q} は任意のスケーリングに対して不変であるため、 $1/s$ を無視できる。 \mathbf{P} について考えてみる。Cayley 変換で表された回転行列である式 (6.33) を式 (6.27) に代入すると、

$$\begin{aligned} \begin{bmatrix} 1 \\ \mathbf{R} \end{bmatrix}^T \text{mat} \left(\mathbf{M} \begin{bmatrix} 1 \\ \mathbf{R} \end{bmatrix} \right) - \text{mat} \left(\mathbf{M} \begin{bmatrix} 1 \\ \mathbf{R} \end{bmatrix} \right)^T \begin{bmatrix} 1 \\ \mathbf{R} \end{bmatrix} &= \frac{1}{s^2} \left(\mathbf{R}^T \text{mat}(\mathbf{M}\mathbf{r}) - \text{mat}(\mathbf{M}\mathbf{r})^T \mathbf{R} \right) \\ &= \frac{1}{s^2} \mathbf{P}. \end{aligned} \quad (6.34)$$

を得る。 $1/s^2 > 0$ であるから、

$$\frac{1}{s^2} \mathbf{P} = \mathbf{P} = \mathbf{0}_{3 \times 3} \quad (6.35)$$

表 6.1: 回転表現の違いによるソルバーの比較

	従来手法			提案手法		
	DLS [26] (Cayley)	OPnP [91] (非単位四元数)	UPnP [36] (四元数)	回転行列	四元数	Cayley
未知数	3	4	4	9	4	3
式の数	3	4	8	27	7	6
最大次数	3	3	3	3	4	4
退化条件	✓					✓
解の個数	27	40	8	40	80	40
elim. templ.	120 × 120	348 × 376	141 × 149	1936 × 1976	630 × 710	124 × 164

となり、スケール定数 s はいかなる値でも影響を与えることはない。これは Q についても同様である。

グレブナー基底生成ツールによると、Cayley 変換による解の個数は 40 個であり、これは OPnP 法や上述した回転行列や単位四元数の半分と同じである。これは Cayley 変換にも関わらず解空間が同一であることを示唆している。また、elimination template のサイズは 124×164 と本節で生成したソルバーの中では最小であった。

Zheng ら [92, 91] が指摘するように、Cayley 変換は単位四元数が $a = 0$ となるときに退化条件となり解が求まらない。しかしながら、Hesch ら [26] が提案するように、前処理としてランダムな回転行列を \mathbf{X}_i にかけて回避可能である。この方法は論文 [26] には掲載されていないが、ソースコード上では実装されている*3。

表現の比較

各回転表現の比較を表 6.1 に示す。処理時間と数値的安定性は elimination template のサイズと解の数に依存する。そのため、提案手法では Cayley 変換を用いることとする。

P と Q の算出において、式 (6.24) に \mathbf{R}^T をかけたことで次数が上がるにも関わらず、いずれの表現も解の数は 40 個（または \pm を含めて 80 個）であることは一見不思議である。しかしながら、 \mathbf{R}^T の乗算はグレブナー基底計算の一部を手作業により未定乗数を消去していると解釈できる。そのため、解の数は変わらないと推測される。

6.4 シミュレーション実験

本節では Cayley 変換を用いた提案手法 (optDLS と呼ぶ) を以下の 7 つ従来手法と比較した評価実験について報告する。

SP+LHM [68] (平面のみ) 大域的最適解への収束を保証する改良版交互最適化法

EPnP+GN [52] (非平面のみ) ガウスニュートン法を併用した高精度版 EPnP 法

SDP [69] 凸最適化に基づく手法

*3 <http://www-users.cs.umn.edu/~joel/>

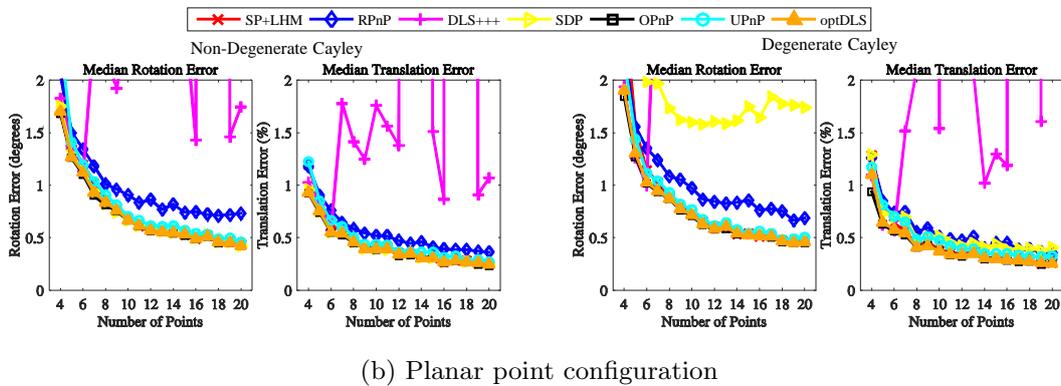
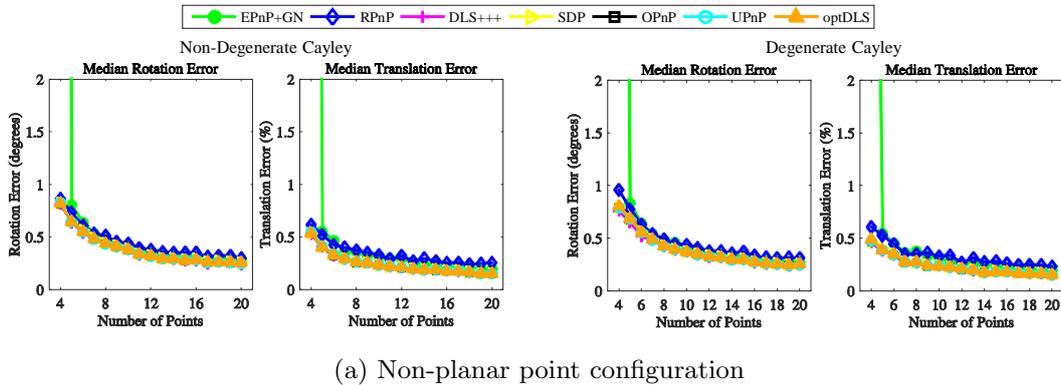


図 6.1: 画像ノイズを $\sigma = 2$ と固定, 点数を $4 \leq n \leq 20$ と変化させたときのノイズ頑健性

RPnP [48] P3P 問題への解法を n 点に拡張した手法

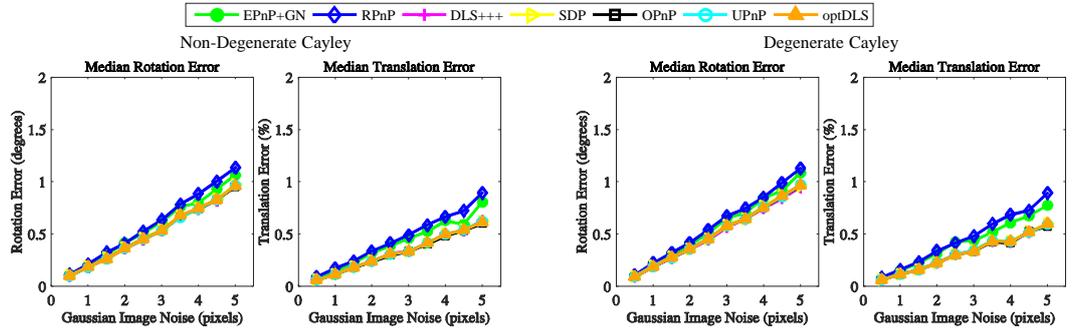
DLS+++ [26, 94] 前処理付き Cayley 変換を用いたグレブナー基底による直接法

OPnP [91] 非単位四元数を用いたグレブナー基底による直接法

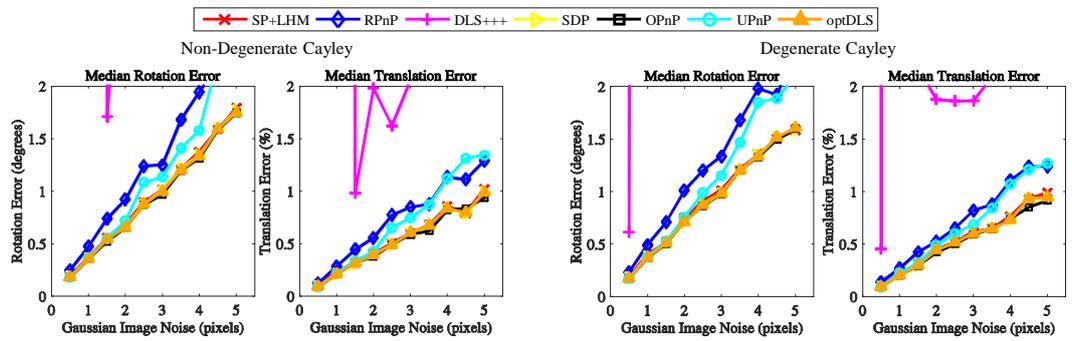
UPnP [36] 単位四元数を用いたグレブナー基底による直接法

DLT 法および非平面を対象とする Lu らの交互最適化法は, 既存研究 [91, 36] で上記の従来手法に及ばないことが示されているので, 本節では評価しない. 提案手法は MATLAB で実装し, C++ 実装の UPnP 以外は OPnP の実装に含まれている MATLAB コードを用いた. 提案手法においては Cayley 変換の退化を避けるために, 前処理としてランダムな回転行列をかけている. また, Zheng ら [92] の実装を参考に, グレブナー基底自動計算ツールにより生成されたコードのうち, `rref()` をバックスラッシュで置き換えて高速化を行った.

本実験では, 画像ノイズを固定して点の数 n を増加させたときの頑健性, 点の数 n を固定して画像ノイズを増加させたときの頑健性, そして処理速度について実験した. 頑健性の実験においては, 点の分布が平面と非平面の 2 種類, および Cayley 変換が退化する場合 ($a = 0$) としない場合 ($a \neq 0$) の 2 種類を構成した. すなわち, 計 4 種類の実験を行った.



(a) Non-planar point configuration



(b) Planar point configuration

図 6.2: 点数を $n = 10$ と固定, 画像ノイズを $0 \leq \sigma \leq 5$ と変化させたときのノイズ頑健性

シミュレーションデータの生成は OPnP のコードに従い以下のように設定した。カメラの解像度は 640×480 ピクセル, 焦点距離は $f = 800$ ピクセルとし, 光学中心は $(c_u, c_v) = (320, 240)$ とする。非平面の場合は, カメラ座標系において x - y - z が $[-2, 2] \times [-2, 2] \times [4, 8]$ の範囲内に 3 次元点群をランダムに生成した。平面の場合は, 世界座標系において x - y - z が $[-2, 2] \times [-2, 2] \times [0, 0]$ の範囲内に 3 次元点群をランダムに生成した。ランダムに回転行列 \mathbf{R}_{true} と並進ベクトル \mathbf{t}_{true} を決定し, 生成した点をカメラに射影した。

回転行列と並進ベクトルの推定誤差は以下のように算出した。

$$e_{rot} = \max_{k \in \{1, 2, 3\}} \cos(\mathbf{r}_{k, \text{true}}^T \mathbf{r}_{k, \text{est}})^{-1} \times 180/\pi \quad [\text{degrees}]$$

$$e_{trans} = \|\mathbf{t}_{\text{true}} - \mathbf{t}_{\text{est}}\| / \|\mathbf{t}_{\text{true}}\| \times 100 \quad [\%]$$
(6.36)

ここで, $\mathbf{r}_{k, \text{true}}$ は $\mathbf{r}_{k, \text{est}}$ は真値 \mathbf{R}_{true} と推定値 \mathbf{R}_{est} の第 k 列である。計測には Core-i3770 を搭載する PC を用いた。

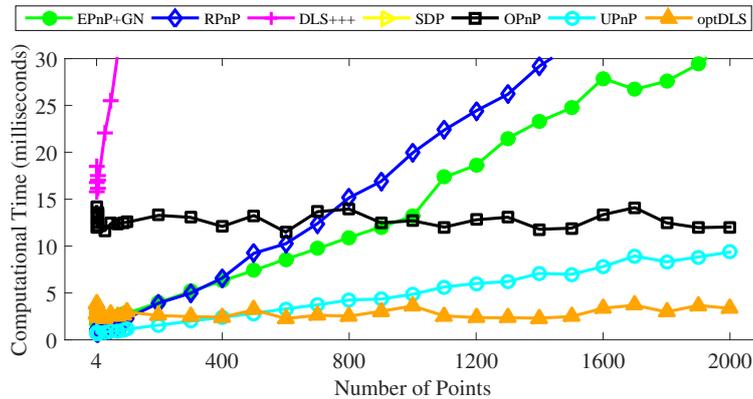


図 6.3: 画像ノイズを $\sigma = 2$ と固定, 点数を $4 \leq n \leq 2000$ と変化させたときの平均実行時間

6.4.1 点数の変化に対するノイズ頑健性

本実験では点の数を $4 \leq n \leq 20$ と変化させた。画像ノイズは、平均ゼロ、標準偏差 $\sigma = 2$ ピクセルとする標準正規分布である。各点数 n について独立に 500 回試行し誤差を計測した。

誤差の中央値を図 6.1 に示す。非平面の場合、ほぼすべての手法が $n \geq 6$ の場合に重なっており、同性能である。RPnP は回転軸の算出、回転角度と並進ベクトルという順で最小二乗法を適用していき全パラメータを使った最適化を行わないため、やや劣る結果となっている。平面の場合は異なった結果が得られた。DLS+++ は前処理としてランダム回転を 3 回行っているが、Cayley 変換の退化条件に関係なく安定性が低い。この原因は、DLS+++ は Cayley 変換のスケーリング $1/\|\mathbf{q}\|$ を無視しているためと考えられる。一方で、式 (6.34)–式 (6.35) で示したように、提案手法 optDLS はスケールに不定な最適性条件に基づいているため、Cayley 変換を用いているが OPnP に匹敵する性能を示している。興味深いことに、UPnP は平面において若干劣っている。この点については次節にてさらに議論する。

6.4.2 画像ノイズの変化に対する頑健性

点の数を $n = 10$ に固定して画像ノイズを $0.5 \leq \sigma \leq 5.0$ と変化させ、各ノイズレベル σ について独立に 500 回試行し誤差を計測した。図 6.2 は誤差の中央値を示している。前節の実験と同様に、提案手法 optDLS と OPnP は同性能である。しかしながら、平面シーンにおいて UPnP は optDLS や OPnP より誤差が大きく、RPnP と同程度である。前節の結果も踏まえると、UPnP は大域的な最適性は持っておらず、RPnP のような準最適手法と思われる。

6.4.3 処理速度

画像ノイズを $\sigma = 2$ ピクセルに固定して点の数を $4 \leq n \leq 2000$ と変化させたときの平均実行時間を図 6.3 に示す。SDP の処理時間は図の範囲内に収まらなかったため記載していない。

提案手法 optDLS の実行時間はどの n に対してもほぼ 3 ミリ秒ほどで、 $n \geq 400$ においては C++ 実装の UPnP を凌駕して最速である。実用上 $n \geq 400$ は AR や VSLAM では珍しくないため、本結果は optDLS がリアルタイム性の高いアプリケーションに適していることを実証している。また、もし C++ 実装の場合はさらなる高速化が期待できる。

提案手法と UPnP の計算時間は $\mathcal{O}(n)$ ではなく $\mathcal{O}(1)$ のように見える。係数行列 M の計算は、式 (6.8) に示すように $\mathcal{O}(n)$ であるが、MATLAB 内の行列積は最適化された並列化処理のため高速に実行される。そのため、処理速度は elimination template の Gauss-Jordan 消去と固有値分解による求解が支配的である。OPnP は四元数の符号不定性を取り除く処理に実行時間の 75% を要しているため、それが不要な optDLS は OPnP よりも 4 倍ほど高速になっている。

6.5 考察

ノイズ頑健性の実験結果 (6.4.1–6.4.2 節) より、提案手法が導出した未定乗数を用いない新たな最適性条件は適切であり、従来不安定であった Cayley 変換に対しても有効であることが分かった。また、MATLAB 実装であることを加味しても処理速度 (6.4.3 節) は UPnP よりも高速であったため、提案手法 optDLS が最も優れた手法と結論づけられる。

6.6 第 6 章のまとめ

本章では、まず代数的誤差に基づく PnP 問題を非線形最適化問題として定式化し、代表的な従来手法の導出と特徴を簡単に紹介した。次に、未定乗数を用いない最適性条件を導出し、異なる回転表現であってもグレンナー基底が利用できることを示した。そして、最もコンパクトな表現である Cayley 変換を用いたソルバーを実装した。最後に、シミュレーションデータを用いた比較評価実験により、提案手法は従来不安定とされた Cayley 変換を用いているにも関わらず、四元数を用いた従来手法と同程度に安定であり、かつ、4 倍以上高速であることを実証した。

本章では、P3P 問題 (第 5 章) と同様に内部パラメータと 3 次元座標にノイズが無いことを仮定しており、その点については議論の余地が残されている。加えて、 n 点すべてが正しい対応点であることを仮定しているが、実際には RANSAC で誤対応点が除去しきれずに、わずかながら誤対応点が混入することもある。このような場合における従来手法及び提案手法の振る舞いについて、今後は詳細な評価実験が必要である。

本章に関連する公表論文は以下の通りである。

中野, 田治米, 野村, 一般カメラモデルの PnP 問題に対するグレンナー基底を用いた統一的解法, 画像の認識・理解シンポジウム (MIRU2011), pp.845–851.

中野, 田治米, 野村, 全停留点の直接計算に基づく一般カメラモデルの PnP 問題に対する統一的解法, 電子情報通信学会論文誌 D, vol.95, no.8, pp.1565–1572, 2012.

Gaku Nakano, Globally Optimal DLS Method for PnP Problem with Cayley parameterization, the 26th British Machine Vision Conference (BMVC2015), 2015.

本章にて提案した手法のコードは以下のウェブサイトにて公開している。

https://github.com/g9nkn/pnp_problem

6.7 付録：グレブナー基底自動計算ツール

Kukelova ら [38] のツール

回転行列

```

% register the generator
setpaths;

% coefficient matrix
M = gbs_Matrix('M%d%d', 9, 9); % 9x9
M = M - tril(M) + triu(M).'; % symmetrization

% rotation matrix
syms a b c d e f g h k
R = [a b c; d e f; g h k];
r = reshape(R.', 9, 1);

% build polynomial equations
matMr = reshape(M * r, 3, 3).';
P      = R.' * matMr - matMr.' * R;
Q      = matMr * R.' - R * matMr.';
g1     = R.'*R - eye(3);
g2     = R*R.' - eye(3);
g3     = R(:,1) - cross(R(:,2),R(:,3));
g4     = R(:,2) - cross(R(:,3),R(:,1));
g5     = R(:,3) - cross(R(:,1),R(:,2));

eqs    = [g1(:,1); g1(2:3,2); g1(3,3);
          g2(:,1); g2(2:3,2); g2(3,3);
          g3; g4; g5;
          P(1,2); P(1,3); P(2,3);
          Q(1,2); Q(1,3); Q(2,3)];

% collect known & unknown variables
unknown = {'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'k'};
vars    = M(:).';
known   = {};
for var = vars
    known = [known {char(var)}];
end

% gbsolver
[res, export] = gbs_CreateCode('solver_optDLS_rotmat', eqs, known, unknown);

```

四元数

```
% register the generator
setpaths;

% coefficient matrix
M = gbs_Matrix('M%d%d', 9, 9); % 9x9
M = M - tril(M) + triu(M).'; % symmetrization

% unit quaternion
syms a b c d
R = [a^2+b^2-c^2-d^2 2*(b*c-a*d) 2*(b*d+a*c)
      2*(b*c+a*d) a^2-b^2+c^2-d^2 2*(c*d-a*b)
      2*(b*d-a*c) 2*(c*d+a*b) a^2-b^2-c^2+d^2];
r = reshape(R.', 9, 1);

% build polynomial equations
matMr = reshape(M * r, 3, 3).';
P      = R.' * matMr - matMr.' * R;
Q      = matMr * R.' - R * matMr.';
g1     = a^2 + b^2 + c^2 + d^2 - 1;
eqs    = [g1;
          P(1,2); P(1,3); P(2,3);
          Q(1,2); Q(1,3); Q(2,3)];

% collect known & unknown variables
unknown = {'a', 'b', 'c', 'd'};
vars    = M(:).';
known   = {};
for var = vars
    known = [known {char(var)}];
end

% gbsolver
[res, export] = gbs_CreateCode('solver_optDLS_quat', eqs, known, unknown);
```

Cayley 変換

```

% register the generator
setpaths;

% coefficient matrix
M = gbs_Matrix('M%d%d', 9, 9); % 9x9
M = M - tril(M) + triu(M).'; % symmetrization

% Cayley parameterization
a = 1; % constant
syms b c d % unknown variables
R = [a^2+b^2-c^2-d^2 2*(b*c-a*d) 2*(b*d+a*c)
      2*(b*c+a*d) a^2-b^2+c^2-d^2 2*(c*d-a*b)
      2*(b*d-a*c) 2*(c*d+a*b) a^2-b^2-c^2+d^2];
r = reshape(R.', 9, 1);

% build polynomial equations
matMr = reshape(M * r, 3, 3).';
P = R.' * matMr - matMr.' * R;
Q = matMr * R.' - R * matMr.';
eqs = [P(1,2); P(1,3); P(2,3);
        Q(1,2); Q(1,3); Q(2,3)];

% collect known & unknown variables
unknown = {'b', 'c', 'd'};
vars = M(:).';
known = {};
for var = vars
    known = [known {char(var)}];
end

% call code generator
[res, export] = gbs_CreateCode('solver_optDLS_cayley', eqs, known, unknown);

```

Larsson ら [42] のツール

回転行列

```
opts = default_options(struct('sparse_template',1));
solv = generate_solver('opt_pnp_nakanoR', @problem_opt_pnp_nakanoR, opts);

function [eqs, data0, eqs_data ] = problem_opt_pnp_nakanoR( data0 )

if nargin < 1 || isempty(data0)
    M = randi(200,9,9);
    M = M+M';
    data0 = M(:);
end

M = reshape(data0,9,9);
xx = create_vars(9);
r = xx;
R = reshape(xx,3,3);

matMr = reshape(M*r,3,3);
P = R'*matMr-matMr'*R;
Q = matMr*R'-R*matMr';
c1 = R'*R-eye(3);
c2 = R*R'-eye(3);

eqs = [P(1,2);P(1,3);P(2,3);
       Q(1,2);Q(1,3);Q(2,3)];

% orthogonality constraints
eqs = [eqs;c1(:);c2(:)];

% positive orientation
eqs = [eqs;R(:,1)-cross(R(:,2),R(:,3))];
eqs = [eqs;R(:,2)-cross(R(:,3),R(:,1))];
eqs = [eqs;R(:,3)-cross(R(:,1),R(:,2))];

if narginout == 3
    xx = create_vars(9+81);
    eqs_data = problem_opt_pnp_nakanoR(xx(10:end));
end
end
```

四元数

```
opts = default_options();
solv = generate_solver('opt_pnp_nakanoQ', @problem_opt_pnp_nakanoQ, opts);

function [eqs, data0, eqs_data] = problem_opt_pnp_nakanoQ( data0 )

if nargin < 1 || isempty(data0)
    M = randi(200,9,9);
    M = M+M';
    data0 = M(:);
end

M = reshape(data0,9,9);
xx = create_vars(4);
a = xx(1);
b = xx(2);
c = xx(3);
d = xx(4);

R = [a^2+b^2-c^2-d^2 2*(b*c-a*d) 2*(b*d+a*c)
      2*(b*c+a*d) a^2-b^2+c^2-d^2 2*(c*d-a*b)
      2*(b*d-a*c) 2*(c*d+a*b) a^2-b^2-c^2+d^2];
r = R(:);
matMr = reshape(M*r,3,3);
P = R'*matMr-matMr'*R;
Q = matMr*R'-R*matMr';

eqs = [P(1,2);P(1,3);P(2,3);
        Q(1,2);Q(1,3);Q(2,3)];
eqs = [eqs; a^2+b^2+c^2+d^2-1];

if nargin == 3
    xx = create_vars(4+81);
    eqs_data = problem_opt_pnp_nakanoQ(xx(5:end));
end
end
```

Cayley 変換

```
opts = default_options();
solv = generate_solver('opt_pnp_nakanoC', @problem_opt_pnp_nakanoc, opts);

function [eqs, data0, eqs_data] = problem_opt_pnp_nakanoC( data0 )

if nargin < 1 || isempty(data0)
    M = randi(200,9,9);
    M = M+M';
    data0 = M(:);
end

M = reshape(data0,9,9);
xx = create_vars(3);
a = 1;
b = xx(1);
c = xx(2);
d = xx(3);

R = [a^2+b^2-c^2-d^2 2*(b*c-a*d) 2*(b*d+a*c);...
     2*(b*c+a*d) a^2-b^2+c^2-d^2 2*(c*d-a*b);...
     2*(b*d-a*c) 2*(c*d+a*b) a^2-b^2-c^2+d^2];
r = R(:);
matMr = reshape(M*r,3,3);
P = R'*matMr-matMr'*R;
Q = matMr*R'-R*matMr';

eqs = [P(1,2);P(1,3);P(2,3);Q(1,2);Q(1,3);Q(2,3)];

if nargout == 3
    xx = create_vars(3+81);
    eqs_data = problem_opt_pnp_nakanoC(xx(4:end));
end
end
```


第7章

PnP、PnPf、PnPfr 問題への統一的な提案手法

本章では PnP 問題を拡張した PnPfr 問題について議論する。PnPfr 問題は PnP 問題に焦点距離とレンズ歪が加わったものであり、PnPf 問題は PnPfr 問題においてレンズ歪をゼロと仮定したときに相当する。まず、レンズ歪に division model を用いて PnPfr 問題の定式化を行う。次に、P4Pf, PnPf, P4Pfr, P5Pfr 問題について代表的な従来手法を簡単に説明する。そして、PnPfr 問題をベースとして、同一アプローチで PnP 問題と PnPf 問題をも解ける手法を2つ提案する。最後に、シミュレーション実験と実画像実験により、従来手法との性能を比較する。

7.1 PnPfr 問題の定式化

PnPf 問題は外部パラメータ（回転行列 R , 並進ベクトル t ）に加えて焦点距離 f を推定する問題である。PnPf 問題の未知パラメータの数は7個のため最小 $n = 4$ 点で解け、P4Pf 問題 [10] と呼ばれる。より正確には3点と4点目の u か v のどちらかを用いれば解けるため、P3.5P 問題 [86] と呼ばれる。

PnPfr 問題はさらにレンズ歪（radial distortion, 放射歪み）が加わる。レンズ歪が k_1 の1つのときは最小 $n = 4$ 点で解ける P4Pfr 問題 [11, 29] と呼ばれ、比較的歪みが小さいレンズに対して用いられる。より広角で歪みが複雑なレンズに対しては、歪みパラメータを3次の項 $\mathbf{k} = [k_1, k_2, k_3]^T$ まで用いて最小 $n = 5$ 点で解ける P5Pfr 問題 [41] と呼ばれる。

PnPfr 問題は PnPf 問題を内包するため、以下では PnPfr 問題を定式化する。光学中心 (c_u, c_v) と歪み中心 $(\tilde{c}_u, \tilde{c}_v)$ は画像の中央 $(w/2, h/2)$ に一致すると仮定して、画像座標は事前に $\tilde{u} \leftarrow \tilde{u} - w/2, \tilde{v} \leftarrow \tilde{v} - h/2$ とシフトしておく。なお、画像座標はレンズ歪みの影響を受けていることを表すために $\tilde{\cdot}$ を付加しており、レンズ歪みモデルは Division model [13, 18] である。

表 7.1: PnPfr 問題への解法の比較.

手法	点数 n	レンズ歪の数	グレンナー 基底	一般性 平面/非平面	解の数
Josephson [29]	4	1	✓	✓	24
Bujnak [11]	4	1	✓		$\begin{cases} 6 & (\text{平面}) \\ 16 & (\text{非平面}) \end{cases}$
Larsson [43]	4	1	✓	✓	
Kukelova [41]	5	≤ 3		✓	8
VPnPfr	≥ 5	≤ 3	✓	✓	20
EDPnPfr	≥ 6	≤ 3	✓		$\begin{cases} 4 & (\text{平面}, n \geq 6) \\ 2 & (\text{非平面}, n = 6) \\ 1 & (\text{非平面}, n \geq 7) \end{cases}$

3次元点 \mathbf{X} からレンズ歪を含む画像平面への射影は次式で表される.

$$\lambda \tilde{\mathbf{m}} = \mathbf{K}(\mathbf{R}\mathbf{X} + \mathbf{t}) \quad \longleftrightarrow \quad \tilde{\mathbf{m}} \propto \mathbf{K}(\mathbf{R}\mathbf{X} + \mathbf{t}) \quad (7.1)$$

ここで,

$$\tilde{\mathbf{m}} = \begin{bmatrix} \tilde{u} \\ \tilde{v} \\ 1 + \mathbf{k}^T \mathbf{d} \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} f & & \\ & f & \\ & & 1 \end{bmatrix} \propto \begin{bmatrix} 1 & & \\ & 1 & \\ & & f^{-1} \end{bmatrix}, \quad (7.2)$$

$$\mathbf{d} = [d, d^2, d^3]^T, \quad d = \tilde{u}^2 + \tilde{v}^2,$$

である. P4Pf 問題は $\mathbf{k} = \mathbf{0}$ として $n = 4$ 点を用いて, P4Pfr 問題は $\mathbf{k} = [k_1, 0, 0]^T$ として $n = 4$ 点を用いて, P5Pfr 問題は $\mathbf{k} = [k_1, k_2, k_3]^T$ として $n = 5$ を用いて式 (7.1) を解く. $n \geq 5$ 点の場合, PnPfr 問題として以下のように定式化される.

$$\begin{aligned} \min_{\mathbf{R}, \mathbf{t}, f, \mathbf{k}} \quad & \sum_{i=1}^n [\tilde{\mathbf{m}}]_{\times} \mathbf{K}(\mathbf{R}\mathbf{X}_i + \mathbf{t}) \\ \text{s.t.} \quad & \mathbf{R}^T \mathbf{R} = \mathbf{I}, \quad \det(\mathbf{R}) = 1 \end{aligned} \quad (7.3)$$

ここで,

$$[\tilde{\mathbf{m}}]_{\times} = \begin{bmatrix} 0 & -(1 + \mathbf{k}^T \mathbf{d}) & \tilde{v} \\ 1 + \mathbf{k}^T \mathbf{d} & 0 & -\tilde{u} \\ -\tilde{v} & \tilde{u} & 0 \end{bmatrix} \quad (7.4)$$

である.

7.2 従来手法

本節では従来提案されてきた PnPf 問題への最小解法と拡張解法, および, PnPfr 問題への最小解法について説明する. なお, PnPfr 問題への従来手法と 7.3–7.4 節で説明する提案手法の特徴を表 7.1 にまとめる.

7.2.1 P4Pf 問題への最小解法

Bujnak らの手法 [10]

Bujnak [10] らは射影的奥行きを計算する P3P 問題に類似した定式化により $n = 4$ から焦点距離 f を求める手法を提案した. 2 点の 3 次元座標間の距離 $d_{ij} = \|\mathbf{X}_i - \mathbf{X}_j\|$ は次式で表される.

$$d_{ij}^2 = \|\mathbf{K}^{-1}(\lambda_i \mathbf{m}_i - \lambda_j \mathbf{m}_j)\|^2 \quad (7.5)$$

ここで, $\mathbf{K}^{-1} = \text{diag}(1, 1, f)$ である. 4 点の中から異なる 2 点の組み合わせ $\{(i, j), (k, l)\}$ を選ぶと, P3P 問題の式 (5.4) に類似した以下の式を得られる.

$$d_{kl}^2 \|\mathbf{K}^{-1}(\lambda_i \mathbf{m}_i - \lambda_j \mathbf{m}_j)\|^2 - d_{ij}^2 \|\mathbf{K}^{-1}(\lambda_k \mathbf{m}_k - \lambda_l \mathbf{m}_l)\|^2 = 0 \quad (7.6)$$

上式は任意の λ について定数倍に不定であるから, $\lambda_1 = 1$ とすると未知数は $\lambda_2, \lambda_3, \lambda_4, f$ の 4 つである. 式 (7.6) の組み合わせは 15 通りがあるが, その中で独立なのは 5 通りである. 未知数に対して 1 式多いため, さらに 5 式の中から 4 式を選び, グレブナー基底自動計算ツール [38] を適用してソルバーを生成する. 結果として 154×180 の elimination template と 16×16 の固有値分解に帰着する.

Wu の手法 [86]

7.1 節で述べたように, P4P 問題の未知数は 7 個であるから, 4 点すべては過剰決定である. Wu [86] はグレブナー基底に基づき 3.5 点で解く手法を提案した. 回転行列表現にオイラー角 (4.4.2) を用いると, z 軸回りの回転 \mathbf{R}_z と内部パラメータ行列 \mathbf{K} を 1 つの行列に統合できる.

$$\begin{aligned} \mathbf{KR} &= (\mathbf{KR}_z)(\mathbf{R}_y \mathbf{R}_x) \\ &= \left(\begin{bmatrix} f & & \\ & f & \\ & & 1 \end{bmatrix} \begin{bmatrix} \cos \theta_z & -\sin \theta_z & 0 \\ \sin \theta_z & \cos \theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) (\mathbf{R}_y \mathbf{R}_x) \\ &= \begin{bmatrix} f_c & -f_s \\ f_s & f_c \\ & & 1 \end{bmatrix} \mathbf{R}_\rho \end{aligned} \quad (7.7)$$

ここで, $f_c = f \cos \theta_z, f_s = f \sin \theta_z$ であり, $\mathbf{R}_\rho = \mathbf{R}_y \mathbf{R}_x$ は四元数 $\mathbf{q} = [1, b, c, 0]$ を用いて表現されている. この表現の特長は $\{f, \theta_z\}$ と $\{-f, \theta_z + \pi\}$ を区別する必要がなくなるため, 解空間を小さくできることである. その後, 変数消去を繰り返し, 最終的に四元数の成分 b と c に関する連立多項式に帰着させる. グレブナー基底により得られる elimination template のサイズは 20×30 と Bujnak らの手法よりも遥かに小さく, 最終的に 10×10 の固有値分解に帰着する. ここまで 3.5 点しか用いていないため, 残り 0.5 点对して得られた解との再投影誤差を計算し, 誤差がしきい値以上の解を取り除くことができる.

7.2.2 PnPf 問題への拡張解法

DLT 法

PnP 問題 (6.2.1) と同様に DLT 法を適用できる。線形化した最適化問題は

$$\begin{aligned} \min_{\mathbf{p}} \quad & \|\mathbf{C}\mathbf{p}\|^2 \\ \text{s.t.} \quad & \|\mathbf{p}\|^2 - 1 = 0 \end{aligned} \quad (7.8)$$

と表させる。係数行列 \mathbf{C} は式 (6.10) と同じであるが、 \mathbf{p} は以下の 3×4 行列

$$\mathbf{P} = \mathbf{K}[\mathbf{R} \ \mathbf{t}] = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \quad (7.9)$$

を 12 次元ベクトル $[p_{11}, \dots, p_{34}]^T$ として表したものである。

内部パラメータ行列 $\mathbf{K} = \text{diag}(f, f, 1)$ と仮定しているので、焦点距離 f は以下のよう
に計算できる*1。まず、 $[p_{31}, p_{32}, p_{33}]$ は回転行列 \mathbf{R} の第 3 行 $[r_{31}, r_{32}, r_{33}]$ と定数倍に
等しいことを利用して、 $\mathbf{p} \leftarrow \mathbf{p} / \|[p_{31}, p_{32}, p_{33}]\|$ と正規化する。次に、 $[p_{11}, p_{12}, p_{13}] =$
 $f[r_{11}, r_{12}, r_{13}]$ であるから、 $f = \|[p_{11}, p_{12}, p_{13}]\|$ として焦点距離が得られる。回転行列
の第 1 行と第 2 行は、それぞれ $[p_{11}, p_{12}, p_{13}]/f$ 、 $[p_{21}, p_{22}, p_{23}]/f$ として計算できるが、
PnP 問題と同様に必ずしも直交制約を満たさない。そこで、式 (4.14) に示した補正処理
を行えばよい。

GPnP 法

Zheng ら [93] は、PnP 問題に対する RPnP 法 [48] を PnPf 問題へと拡張した。まず、
全 n 点の 3 次元点を、ある 2 点を共有する 3 点ずつの組に分割し、その 2 点を中心とす
る中間座標系において P4Pf 問題の式 (7.5) を $n - 2$ 本立てる。そして、 $n - 2$ 本の式を
最小二乗法により解いて 2 点への射影的奥行きと焦点距離 f を算出し、中間座標系におけ
る回転軸を決定する。最後に、RPnP 法と同様に DLT 法に似たアプローチで回転角度と
並進ベクトルを算出する。

EPnPfr 法

Kanaeva ら [30] は PnP 問題に対する EPnP 法 [46, 52] を PnPf 問題へと拡張した*2。
EPnP 法において 4 つの制御点を算出する制約式を、焦点距離 f を含む線形最小二乗法
として定式化する。EPnP 法との違いは、DLT 法ではなく正則化付き最小自乗問題とし
て解くことである。安定的な収束のためには正則化問題のパラメータを細かく制御しなく
てはならず、論文 [30] では経験的な値が 2 パターン（高速版、厳密版）掲載されている。

*1 カメラキャリブレーションのようにせん断係数 γ や光学中心 (c_u, c_v) を計算する場合は QR 分解を利用
する。詳細は文献 [24] の 6.2.4 節を参照されたい

*2 名称に "fr" がついているが、PnPfr 問題への解法ではない。語源について論文中では触れられていない
が、手法の特徴から "focal length" と "Regularization" の頭文字と推測される。

7.2.3 P4Pfr, P5Pfr 問題への最小解法

Josephson らの手法 [29]

Josephson らは P4Pfr 問題への解法を初めて提案した。レンズ歪を含む射影式 (7.1) にそのままグレブナー基底を利用する素朴な方法である。四元数 $\mathbf{q} = [a, b, c, d]^T$ を用いて回転行列 \mathbf{R} を表すと、4 点 $i = \{1, 2, 3, 4\}$ について以下の式が成り立つ。

$$\begin{aligned} [\tilde{\mathbf{m}}_i]_{\times} \mathbf{K}(\mathbf{R} \mathbf{X}_i + \mathbf{t}) &= 0 \\ \|\mathbf{q}\|^2 - 1 &= 0 \end{aligned} \quad (7.10)$$

連立方程式を単純化する工夫として、P3P 問題の直接解法のように \mathbf{X}_1 を中心とする座標系に変換する。手作業によるグレブナー基底計算の結果、elimination template のサイズは 1134×720 であり、解の数は 24 個である。平面と非平面を区別する必要はないが、elimination template が大きいため RANSAC には不向きである。

Bujnak らの手法 [11]

Bujnak らは、平面と非平面によって場合分けをすることで効率的な P4Pfr 解法を提案した。そして、グレブナー基底自動計算ツール [38] を利用して、平面の場合は 12×18 、非平面の場合は 136×152 の elimination template を算出した。

Larsson らの手法 [43]

Larsson らは、平面シーンに対する P4Pfr 問題のグレブナー基底を詳細に解析し、平面と非平面の両方に対応可能な単項式基底 (monomial basis) を発見した。それにより、Bujnak らが平面と非平面に対して別々に計算していたグレブナー基底を統一し、場合分けの不要な P4Pfr 解法を提案した。また、射影行列 \mathbf{P} の左 3×3 行列が \mathbf{K} と \mathbf{R} に分解可能になる制約条件を新たに導入し、elimination template を 40×50 にまで縮小できることを示した。

Kukelova らの手法 [41]

Kukelova らは、 $n = 5$ 点を用いれば平面と非平面を区別することなく P5Pfr 問題が解けることを明らかにした。行列 $[\mathbf{m}]_{\times}$ の第 3 行 $[-\tilde{v}, \tilde{u}, 0]$ に歪パラメータ \mathbf{k} が含まれないことに着目する。式 (7.10) の第 3 行目のみを取り出すと次式を得る。

$$\begin{bmatrix} -\tilde{v}_1 \mathbf{X}_1^T & \tilde{u}_1 \mathbf{X}_1^T \\ \vdots & \vdots \\ -\tilde{v}_5 \mathbf{X}_5^T & \tilde{u}_5 \mathbf{X}_5^T \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \end{bmatrix} = \mathbf{M}\mathbf{v} = \mathbf{0}_{5 \times 1} \quad (7.11)$$

ただし、 \mathbf{p}_i は $\mathbf{P} = \mathbf{K}[\mathbf{R} \ \mathbf{t}]$ の第 i 行を縦ベクトルで表したものである。係数行列 \mathbf{M} は 5×8 であるから、 $\mathbf{v} = [\mathbf{p}_1, \mathbf{p}_2]$ は \mathbf{M} の零空間ベクトル $\mathbf{n}_i, i \in \{1, 2, 3\}$ の線形結合により

$$\mathbf{v} \propto \sum_{i=1}^3 \alpha_i \mathbf{n}_i \quad (7.12)$$

として表せる。そして $[p_{11}, p_{12}, p_{13}]$ と $[p_{21}, p_{22}, p_{23}]$ の直交制約を用いると、1つの α_i についての4次式に帰着する。そして、 $[\mathbf{m}]_{\times}$ の第1行もしくは第2行を使って、残りのパラメータである並進ベクトル \mathbf{t} 、焦点距離 f 、レンズ歪 \mathbf{k} を算出する。グレンナー基底を用いることなく1変数4次式を解くだけなので、Bujnak らの P4Pfr 解法よりも100倍以上高速である。 $n = 5$ のため P4Pfr 解法よりも RASAC に要する反復数は多いが、RANSAC 全体の計算時間は P4Pfr 解法よりも数倍速い。

7.3 提案手法 1: VPnPfr 法

本節では、PnPfr 問題だけでなく包含される PnP 問題と PnPf 問題をも解く統一的な手法を提案する。提案手法は PnPfr 問題を2つの部分問題に分割し、第1の部分問題はグレンナー基底を、第2の部分問題は線形最小二乗法を用いて解く。そして、得られた解を初期値として原問題の KKT 条件を満たすように root polishin を行い、解の精度を向上させる。3つの問題に対する汎用的 (Versatile) な手法という意味で VPnPfr と名付ける。

部分問題の定義について概要を次節にて説明し、部分問題の解法を 7.3.2–7.3.4 節にて詳細に説明する。

7.3.1 概要

式 (7.4) に示す $[\mathbf{m}_i]_{\times}$ の第1列、第2列、第3列を、それぞれ $\mathbf{a}_i^T, \mathbf{b}_i^T, \mathbf{c}_i^T$ とする。これらを用いると、PnPfr 問題 (式 (7.3)) の目的関数は次式で表される。

$$\min_{\mathbf{R}, \mathbf{t}, f, \mathbf{k}} \sum_{i=1}^n \left\| [\mathbf{m}_i]_{\times} \mathbf{q}_i \right\|^2 = \sum_{i=1}^n (\mathbf{a}_i^T \mathbf{q}_i)^2 + (\mathbf{b}_i^T \mathbf{q}_i)^2 + (\mathbf{c}_i^T \mathbf{q}_i)^2, \quad (7.13)$$

ただし、 $\mathbf{q}_i = \mathbf{K}(\mathbf{R}\mathbf{X}_i + \mathbf{t})$ とおいた。

4.7 節で説明したように、式 (7.13) は代数的誤差であるから、ノイズがなければ \mathbf{m}_i と \mathbf{q}_i は共線である。言い換えると、式 (7.13) の最小化は、ノイズを含む $\mathbf{m}_i \times \mathbf{q}_i$ のすべての成分をゼロに近づける最適なパラメータを探索することに等しい。それゆえ、それぞれの成分を独立した部分問題とみなして最適化した場合、いずれかの解は原問題の大域的最適解に近いことが期待される。

それでは部分問題を定義する．3つの項 $\mathbf{a}_i^T \mathbf{q}_i$, $\mathbf{b}_i^T \mathbf{q}_i$, $\mathbf{c}_i^T \mathbf{q}_i$ を展開すると,

$$\mathbf{a}_i^T \mathbf{q}_i = -(1 + \mathbf{k}^T \mathbf{d})(\mathbf{r}_2^T \mathbf{X}_i + t_y) + v_i f^{-1}(\mathbf{r}_3^T \mathbf{X}_i + t_z) \quad (7.14)$$

$$\mathbf{b}_i^T \mathbf{q}_i = (1 + \mathbf{k}^T \mathbf{d})(\mathbf{r}_1^T \mathbf{X}_i + t_x) - u_i f^{-1}(\mathbf{r}_3^T \mathbf{X}_i + t_z) \quad (7.15)$$

$$\mathbf{c}_i^T \mathbf{q}_i = -v_i(\mathbf{r}_1^T \mathbf{X}_i + t_x) + u_i(\mathbf{r}_2^T \mathbf{X}_i + t_y) \quad (7.16)$$

となる．ここで, \mathbf{r}_j は回転行列 \mathbf{R} の第 j 行を表す．Kukelova らの手法 (7.2.3) でも説明したように, 式 (7.16) には歪パラメータ \mathbf{k} が現れず, 外部パラメータの一部 $\mathbf{r}_1, \mathbf{r}_2, t_x, t_y$ で構成されている．一方, 式 (7.14) と式 (7.15) はすべての未知数を含んでいる．

そこで, 第 1 の部分問題を以下のように定義する．

$$\begin{aligned} \min_{\mathbf{r}_1, \mathbf{r}_2, t_x, t_y} \quad & \sum_{i=1}^n (\mathbf{c}_i^T \mathbf{q}_i)^2 \\ \text{s.t.} \quad & \|\mathbf{r}_1\|^2 = 1, \quad \|\mathbf{r}_1\|^2 - \|\mathbf{r}_2\|^2 = 0, \quad \mathbf{r}_1^T \mathbf{r}_2 = 0. \end{aligned} \quad (7.17)$$

式 (7.17) の変数は 8 個であるが, \mathbf{r}_1 と \mathbf{r}_2 に関する制約条件のため実際の自由度は 5 である．すなわち, $n \geq 5$ 点が与えられれば解ける．式 (7.17) から \mathbf{r}_1 と \mathbf{r}_2 が得られれば, 回転行列 \mathbf{R} の第 3 行は $\mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2$ として計算できる．

得られた \mathbf{R}, t_x, t_y を式 (7.14)–(7.15) に代入すると, 残る未知数は t_z, f, \mathbf{k} の 5 つである．すでに回転行列は計算されているため, 残りのパラメータについては無制約となる．

それゆえ, 第 2 の部分問題は以下のように表せる．

$$\min_{t_z, f, \mathbf{k}} \sum_{i=1}^n (\mathbf{a}_i^T \mathbf{q}_i)^2 + (\mathbf{b}_i^T \mathbf{q}_i)^2. \quad (7.18)$$

式 (7.18) は 1 つの対応点から 2 つの拘束式が得られるため, 第 1 の部分問題の入力が $n \geq 5$ 点であれば式 (7.18) は解ける．

上述した第 1 の部分問題と第 2 の部分問題の解は, 式 (7.13) で表した原問題の最適解ではない．そこで, root polishing を実行して最終的な精度と最適性を向上させる．

部分問題それぞれの具体的な解法について 7.3.2–7.3.4 節で説明する．

7.3.2 グレブナー基底による部分問題 1 の解法

式 (7.17) の目的関数は, 式 (7.16) を用いて以下のように書き換えられる．

$$\min_{\mathbf{r}_1, \mathbf{r}_2, t_x, t_y} \sum_{i=1}^n (\mathbf{c}_i^T \mathbf{q}_i)^2 = \|\mathbf{A}\hat{\mathbf{r}} + \mathbf{B}\hat{\mathbf{t}}\|^2, \quad (7.19)$$

ここで,

$$\mathbf{A} = \begin{bmatrix} -v_1 \mathbf{X}_1^T & u_1 \mathbf{X}_1^T \\ \vdots & \vdots \\ -v_n \mathbf{X}_n^T & u_n \mathbf{X}_n^T \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} -v_1 & u_1 \\ \vdots & \vdots \\ -v_n & u_n \end{bmatrix}, \quad \hat{\mathbf{r}} = \begin{bmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \end{bmatrix}, \quad \hat{\mathbf{t}} = \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (7.20)$$

である。並進ベクトルの x, y 成分である $\hat{\mathbf{t}}$ については無制約であるから、 $\hat{\mathbf{t}}$ は $\hat{\mathbf{r}}$ の関数として以下のように表される*3。

$$\hat{\mathbf{t}} = -(\mathbf{B}^T \mathbf{B})^{-1} \mathbf{B}^T \mathbf{A} \hat{\mathbf{r}}. \quad (7.21)$$

上式を式 (7.19) に代入すると、新たな制約付き最適化問題を次式のように定義できる。

$$\begin{aligned} \min_{\mathbf{r}_1, \mathbf{r}_2} \quad & \hat{\mathbf{r}}^T \mathbf{M} \hat{\mathbf{r}} \\ \text{s.t.} \quad & \|\mathbf{r}_1\|^2 = 1, \quad \|\mathbf{r}_1\|^2 - \|\mathbf{r}_2\|^2 = 0, \quad \mathbf{r}_1^T \mathbf{r}_2 = 0, \end{aligned} \quad (7.22)$$

ただし、

$$\mathbf{M} = \mathbf{A}^T \mathbf{A} - \mathbf{A}^T \mathbf{B} (\mathbf{B}^T \mathbf{B})^{-1} \mathbf{B}^T \mathbf{A} \quad (7.23)$$

である。

式 (7.22) は PnP 問題 (第6章) に類似した最適化問題である。そのため、同様にグレブナー基底を用いて KKT 条件の連立多項式を解けば、大域的に最適な $\hat{\mathbf{r}}$ を求めることができる。PnP 問題 [36, 54, 91] のように回転行列を四元数で表現し、グレブナー基底計算ツール [38, 42] を利用すると、PnP 問題と同じ 40 個の解を持つことが分かる。しかしながら、式 (7.22) において $\hat{\mathbf{r}}$ には符号の不定性が存在するため、 $-\hat{\mathbf{r}}$ も制約を満たす最小値を与える。すなわち、実際の解の個数は 40 個ではなく 20 個である。四元数や Cayley 変換は $\pm \hat{\mathbf{r}}$ を区別できないため、本節では \mathbf{r}_1 と \mathbf{r}_2 を用いて解を 20 個に抑えるグレブナー基底を導出する。

6 × 6 の対称行列 \mathbf{M} を、3 × 3 の区分行列 M_{ij} を用いて

$$\mathbf{M} = \begin{bmatrix} M_{11} & M_{12} \\ M_{12}^T & M_{22} \end{bmatrix} \quad (7.24)$$

と表す。式 (7.22) のラグランジュ関数は

$$\begin{aligned} L = & \mathbf{r}_1^T M_{11} \mathbf{r}_1 + 2\mathbf{r}_1^T M_{12} \mathbf{r}_2 + \mathbf{r}_2^T M_{22} \mathbf{r}_2 \\ & + \lambda_1 (1 - \|\mathbf{r}_1\|^2) + \lambda_2 (\|\mathbf{r}_1\|^2 - \|\mathbf{r}_2\|^2) + 2\lambda_3 \mathbf{r}_1^T \mathbf{r}_2, \end{aligned} \quad (7.25)$$

と書ける。ここで、 $\lambda_1, \lambda_2, \lambda_3$ は未定定数であり、 λ_3 への乗数 2 は便宜上の係数である。式 (7.25) の 1 次最適性条件は以下のように表せる。

$$\frac{\partial L}{\partial \mathbf{r}_1} = M_{11} \mathbf{r}_1 + M_{12} \mathbf{r}_2 + (\lambda_2 - \lambda_1) \mathbf{r}_1 + \lambda_3 \mathbf{r}_2 = \mathbf{0}_{3 \times 1} \quad (7.26)$$

$$\frac{\partial L}{\partial \mathbf{r}_2} = M_{22} \mathbf{r}_2 + M_{12}^T \mathbf{r}_1 - \lambda_2 \mathbf{r}_2 + \lambda_3 \mathbf{r}_1 = \mathbf{0}_{3 \times 1} \quad (7.27)$$

$$\frac{\partial L}{\partial \lambda_1} = 1 - \|\mathbf{r}_1\|^2 = 0 \quad (7.28)$$

$$\frac{\partial L}{\partial \lambda_2} = \|\mathbf{r}_1\|^2 - \|\mathbf{r}_2\|^2 = 0 \quad (7.29)$$

$$\frac{\partial L}{\partial \lambda_3} = \mathbf{r}_1^T \mathbf{r}_2 = 0 \quad (7.30)$$

*3 導出は 6.1 節の PnP 問題と同様のため省略する。

次に, $[\mathbf{r}_1]_{\times}$ と $[\mathbf{r}_2]_{\times}$ を式 (7.26) と式 (7.27) にそれぞれ掛けると, 次式を得る.

$$[\mathbf{r}_1]_{\times} (\mathbf{M}_{11}\mathbf{r}_1 + \mathbf{M}_{12}\mathbf{r}_2) + \lambda_3 [\mathbf{r}_1]_{\times} \mathbf{r}_2 = \mathbf{0} \quad (7.31)$$

$$[\mathbf{r}_2]_{\times} (\mathbf{M}_{22}\mathbf{r}_2 + \mathbf{M}_{12}^T\mathbf{r}_1) + \lambda_3 [\mathbf{r}_2]_{\times} \mathbf{r}_1 = \mathbf{0} \quad (7.32)$$

クロス積の性質より $[\mathbf{r}_1]_{\times} \mathbf{r}_2 = -[\mathbf{r}_2]_{\times} \mathbf{r}_1$ であるから, 式 (7.31) と式 (7.32) を足せば, λ_3 を消去できる.

$$[\mathbf{r}_1]_{\times} (\mathbf{M}_{11}\mathbf{r}_1 + \mathbf{M}_{12}\mathbf{r}_2) + [\mathbf{r}_2]_{\times} (\mathbf{M}_{22}\mathbf{r}_2 + \mathbf{M}_{12}^T\mathbf{r}_1) = \mathbf{0} \quad (7.33)$$

さらに, \mathbf{r}_2^T を式 (7.26) に, \mathbf{r}_1^T を式 (7.27) に掛けると, 次式を得る.

$$\mathbf{r}_2^T (\mathbf{M}_{11}\mathbf{r}_1 + \mathbf{M}_{12}\mathbf{r}_2) + \lambda_3 \|\mathbf{r}_2\|^2 = 0 \quad (7.34)$$

$$\mathbf{r}_1^T (\mathbf{M}_{22}\mathbf{r}_2 + \mathbf{M}_{12}^T\mathbf{r}_1) + \lambda_3 \|\mathbf{r}_1\|^2 = 0 \quad (7.35)$$

式 (7.29) に示すように \mathbf{r}_1 と \mathbf{r}_2 の L2 ノルムは互いに等しいから, 式 (7.34) から式 (7.35) を引けば λ_3 を消去できる.

$$\mathbf{r}_2^T (\mathbf{M}_{11}\mathbf{r}_1 + \mathbf{M}_{12}\mathbf{r}_2) - \mathbf{r}_1^T (\mathbf{M}_{22}\mathbf{r}_2 + \mathbf{M}_{12}^T\mathbf{r}_1) = 0. \quad (7.36)$$

ここで着目すべきは, 式 (7.29)–(7.30) の制約条件が満たされていれば, 式 (7.33) と式 (7.36) は \mathbf{r}_1 の正規化に関係なく成立することである. つまり, $\hat{\mathbf{r}}$ の符号不定性を除去するために, $\|\mathbf{r}_1\|^2 = 1$ ではなく $r_{11} = 1$ や $r_{11} + r_{12} + r_{13} = 1$ のような線形制約を用いることができる. それゆえ, 式 (7.29), 式 (7.30), 式 (7.33), 式 (7.36) に新たな線形制約を加えた連立方程式を解けば, \mathbf{r}_1 と \mathbf{r}_2 が算出できる.

Kukelova らのグレブナー基底自動計算ツール [38] を用いると, $97 \times 117^{*4}$ の elimination template と 20×20 の固有値分解に帰着する. 最大で 20 個の実数解 (\mathbf{r}_1 と \mathbf{r}_2) が得られ, 符号不定性を考慮すると 1 つの解から 2 つの回転行列が復元できる.

$$\mathbf{R} = \begin{bmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \\ (\mathbf{r}_1 \times \mathbf{r}_2)^T \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} -\mathbf{r}_1^T \\ -\mathbf{r}_2^T \\ (\mathbf{r}_1 \times \mathbf{r}_2)^T \end{bmatrix}. \quad (7.37)$$

また, 並進ベクトルの x, y 成分 $\hat{\mathbf{t}}$ は式 (7.21) により算出できる. グレブナー基底自動計算ツールを利用するためのコードは章末の付録 7.8 に記載している.

7.3.3 部分問題 2 の解法

第 2 の部分問題は, PnP 問題か PnPf 問題か PnPfr 問題かにより解法が少しずつ異なる. 本節では, まず PnPfr 問題に対する解法について述べ, 次に包含される PnPf 問題と PnP 問題について順に説明する.

*4 ツールのコードを一部変更し, 次数の大きさによる式の並び替えを実行した場合のサイズ. デフォルトでは 105×125 である.

PnPfr 問題

前節で得られたパラメータ \mathbf{R} , t_x , t_y を既知とすると, 式 (7.18) は以下のように書き換えられる.

$$\min_{t_z, f, \mathbf{k}} \sum_{i=1}^n \left\| \begin{bmatrix} \mathbf{a}_i^T \\ \mathbf{b}_i^T \end{bmatrix} \mathbf{q}_i \right\|^2 = \|\mathbf{L}\mathbf{x} + \mathbf{g}\|^2, \quad (7.38)$$

ここで,

$$\mathbf{L} = \begin{bmatrix} v_1 & v_1 z_1^c & -y_1^c \mathbf{d}_1^T \\ -u_1 & -u_1 z_1^c & x_1^c \mathbf{d}_1^T \\ & \vdots & \\ v_n & v_n z_1^c & -y_n^c \mathbf{d}_n^T \\ -u_n & -u_n z_1^c & x_n^c \mathbf{d}_n^T \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} f^{-1} t_z \\ f^{-1} \\ \mathbf{k} \end{bmatrix}, \quad \mathbf{g} = \begin{bmatrix} -y_1^c \\ x_1^c \\ \vdots \\ -y_n^c \\ x_n^c \end{bmatrix}, \quad (7.39)$$

$$x_i^c = \mathbf{r}_1^T \mathbf{X}_i + t_x, \quad y_i^c = \mathbf{r}_2^T \mathbf{X}_i + t_y, \quad z_i^c = \mathbf{r}_3^T \mathbf{X}_i.$$

である. 上式は未知パラメータをまとめたベクトル \mathbf{x} についての線形最小二乗法であるから, 正規方程式 $\mathbf{x} = -(\mathbf{L}^T \mathbf{L})^{-1} \mathbf{L}^T \mathbf{g}$ を解けば解が得られる. そして, f は \mathbf{x} の第2成分, \mathbf{k} は第3成分から第6成分, t_z は \mathbf{x} の第1の成分を第2の成分で割った値として復元できる.

PnPf 問題

PnPf 問題では歪パラメータ $\mathbf{k} = \mathbf{0}$ であるから, \mathbf{L} と \mathbf{x} は以下のように書き換えられる.

$$\mathbf{L} = \begin{bmatrix} v_1 & v_1 z_1^c \\ -u_1 & -u_1 z_1^c \\ \vdots & \vdots \\ v_n & v_n z_1^c \\ -u_n & -u_n z_1^c \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} f^{-1} t_z \\ f^{-1} \end{bmatrix} \quad (7.40)$$

また, \mathbf{g} は式 (7.39) と同じである.

PnP 問題

PnP 問題ではさらに焦点距離 f が既知であるから, 次式のように書ける.

$$\mathbf{L} = f^{-1} \begin{bmatrix} v_1 \\ -u_1 \\ \vdots \\ v_n \\ -u_n \end{bmatrix}, \quad \mathbf{x} = t_z, \quad \mathbf{g} = f^{-1} \begin{bmatrix} v_1 z_1^c \\ -u_1 z_1^c \\ \vdots \\ v_n z_1^c \\ -u_n z_1^c \end{bmatrix} + \begin{bmatrix} -y_1^c \\ x_1^c \\ \vdots \\ -y_n^c \\ x_n^c \end{bmatrix} \quad (7.41)$$

正規化画像座標 $\mathbf{K}^{-1} \mathbf{m}$ を用いた場合は, $f = 1$ とすればよい.

7.3.4 Root polishing

7.3.2–7.3.3 節の結果として、すべての未知数を推定できる。しかしながら、算出されたパラメータは部分問題の解であって原問題の最適解ではない。解の精度と最適性を向上するために、本節で述べる root polishing を実行して 1 次最適性条件を厳密に満たすように解を補正する。前節と同様に、まず PnPfr 問題の場合について説明し、次に PnPfr 問題と PnP 問題の場合について記述する。

PnPfr 問題

PnPfr 問題の原問題 (式 (7.13)) において、 \mathbf{c}_i は \mathbf{a}_i と \mathbf{b}_i に対して線形従属であるから、目的関数を以下のように等価に書き換えられる。

$$\min_{\mathbf{r}, \mathbf{t}, f, \mathbf{k}} \sum_{i=1}^n (\mathbf{a}_i^T \mathbf{q}_i)^2 + (\mathbf{b}_i^T \mathbf{q}_i)^2 = \|\mathbf{C}_{(f, \mathbf{k})} \mathbf{r} + \mathbf{D}_{(f, \mathbf{k})} \mathbf{t}\|^2 \quad (7.42)$$

ここで、 $\mathbf{C}_{(f, \mathbf{k})}$ と $\mathbf{D}_{(f, \mathbf{k})}$ は、 f と \mathbf{k} を含むそれぞれ $n \times 9$ と $n \times 3$ の係数行列であり、次式で表される。

$$\begin{aligned} \mathbf{C}_{(f, \mathbf{k})} &= \mathbf{A}_1 \mathbf{W} + \mathbf{A}_2 \mathbf{Y} \\ \mathbf{D}_{(f, \mathbf{k})} &= \mathbf{B}_1 \mathbf{K} + \mathbf{B}_2 \mathbf{Z} \end{aligned} \quad (7.43)$$

ここで、クロネッカー積 \otimes を用いて、

$$\begin{aligned} \mathbf{A}_1 &= \begin{bmatrix} & -\mathbf{p}_1^T & v_1 \mathbf{p}_1^T \\ \mathbf{0}_{n \times 3} & \vdots & \vdots \\ & -\mathbf{p}_n^T & v_n \mathbf{p}_n^T \\ \mathbf{p}_1^T & & -u_1 \mathbf{p}_1^T \\ \vdots & \mathbf{0}_{n \times 3} & \vdots \\ \mathbf{p}_n^T & & -u_n \mathbf{p}_n^T \end{bmatrix}, \quad \mathbf{A}_2 = \begin{bmatrix} & -(\mathbf{p}_1 \otimes \mathbf{d}_1)^T \\ \mathbf{0}_{n \times 9} & \vdots \\ & -(\mathbf{p}_n \otimes \mathbf{d}_n)^T \\ (\mathbf{p}_1 \otimes \mathbf{d}_1)^T & \\ \vdots & \mathbf{0}_{n \times 9} \\ (\mathbf{p}_n \otimes \mathbf{d}_n)^T & \end{bmatrix}, \\ \mathbf{B}_1 &= \begin{bmatrix} & -1 & v_1 \\ \mathbf{0}_{n \times 1} & \vdots & \vdots \\ & -1 & v_n \\ 1 & & u_1 \\ \vdots & \mathbf{0}_{n \times 1} & \vdots \\ 1 & & u_n \end{bmatrix}, \quad \mathbf{B}_2 = \begin{bmatrix} & -\mathbf{d}_1^T \\ \mathbf{0}_{n \times 3} & \vdots \\ & -\mathbf{d}_n^T \\ \mathbf{d}_1^T & \\ \vdots & \mathbf{0}_{n \times 3} \\ \mathbf{d}_n^T & \end{bmatrix}, \\ \mathbf{W} &= \begin{bmatrix} \mathbf{I}_{6 \times 6} & \\ & f^{-1} \mathbf{I}_{3 \times 3} \end{bmatrix}, \quad \mathbf{Y} = [\mathbf{I}_6 \otimes \mathbf{k}, \quad \mathbf{0}_{18 \times 3}], \quad \mathbf{Z} = [\mathbf{I}_2 \otimes \mathbf{k}, \quad \mathbf{0}_{6 \times 1}]. \end{aligned} \quad (7.44)$$

である。

6.1 節の PnP 問題で導出したように、並進ベクトル \mathbf{t} は他のパラメータの関数として

$$\mathbf{t}_{(\mathbf{r}, f, \mathbf{k})} = -(\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T \mathbf{C} \mathbf{r}. \quad (7.45)$$

と書ける．そして，式 (7.45) を式 (7.42) に代入すると，PnPfr 問題は以下の制約付き最適化問題となる．

$$\begin{aligned} \min_{\mathbf{r}, f, \mathbf{k}} \quad & \mathbf{r}^T \mathbf{G}_{(f, \mathbf{k})} \mathbf{r} \\ \text{s.t.} \quad & \mathbf{R}^T \mathbf{R} = \mathbf{I}, \quad \det(\mathbf{R}) = 1 \end{aligned} \quad (7.46)$$

ただし，

$$\mathbf{G}_{(f, \mathbf{k})} = \mathbf{C}^T \mathbf{C} - \mathbf{C}^T \mathbf{D} (\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T \mathbf{C} \quad (7.47)$$

である．

Root polishing は，前節で得られた部分問題 1 と部分問題 2 の解を初期として式 (7.46) を最適化すればよい．ここでオイラー角 (4.4.2 節) や Cayley 変換 (4.4.4 節) を用いて無制約最適化を行うと，数値的な退化条件に陥る可能性がある．そこで，6.3.1 節で導出した回転行列の表現に依存しない KKT 条件を最適化する．すなわち，下記の連立方程式を用いて最適化を行えばよい．

$$\begin{aligned} \mathbf{R}^T \text{mat}(\mathbf{G}\mathbf{r}) - \text{mat}(\mathbf{G}\mathbf{r})^T \mathbf{R} &= \mathbf{0}_{3 \times 3} \\ \text{mat}(\mathbf{G}\mathbf{r}) \mathbf{R}^T - \mathbf{R} \text{mat}(\mathbf{G}\mathbf{r})^T &= \mathbf{0}_{3 \times 3} \\ \mathbf{R}^T \mathbf{R} - \mathbf{I} &= \mathbf{0}_{3 \times 3} \\ \det(\mathbf{R}) - 1 &= 0 \\ \frac{\partial}{\partial f} \mathbf{r}^T \mathbf{G}\mathbf{r} &= 0 \\ \frac{\partial}{\partial \mathbf{k}} \mathbf{r}^T \mathbf{G}\mathbf{r} &= \mathbf{0}_{3 \times 1} \end{aligned} \quad (7.48)$$

ここで， $\text{mat}(\cdot)$ は 9 次元ベクトルを 3×3 の正方行列へ変形するオペレータである．6.3.2 節で証明したように，式 (7.48) はどのような回転行列の表現を用いることもできる．そこで，任意の回転表現を用いて式 (7.48) に対してガウスニュートン法を適用すればよい．数回から 10 回程度の反復で十分である．ガウスニュートン法の収束後，式 (7.45) に従って \mathbf{t} を再計算すればすべてのパラメータが得られる．

PnPf 問題

PnPf 問題では $\mathbf{k} = \mathbf{0}$ であるから，式 (7.43) において \mathbf{Y} と \mathbf{Z} はゼロ行列である．ゆえに，

$$\begin{aligned} \mathbf{C}_{(f)} &= \mathbf{A}_1 \mathbf{W} \\ \mathbf{D}_{(f)} &= \mathbf{B}_1 \mathbf{K} \end{aligned} \quad (7.49)$$

である．上式を式 (7.45) と式 (7.47) に代入すると，並進ベクトルと目的関数の係数行列は

$$\begin{aligned} \mathbf{t} &= -\mathbf{K}^{-1} (\mathbf{B}_1^T \mathbf{B}_1)^{-1} \mathbf{B}_1^T \mathbf{A}_1 \mathbf{W} \mathbf{r} \\ \mathbf{G} &= \mathbf{W} (\mathbf{A}_1^T \mathbf{A}_1 - \mathbf{A}_1^T \mathbf{B}_1 (\mathbf{B}_1^T \mathbf{B}_1)^{-1} \mathbf{B}_1^T \mathbf{A}_1) \mathbf{W} \end{aligned} \quad (7.50)$$

と表される．

PnP 問題

PnP 問題では焦点距離 f が既知であるから、その値を式 (7.50) に代入すればよい。もしくは、正規化画像座標 $K^{-1}\mathbf{m}$ を用いた場合は、 $f = 1$ であるから

$$\begin{aligned} \mathbf{t} &= -(\mathbf{B}_1^\top \mathbf{B}_1)^{-1} \mathbf{B}_1^\top \mathbf{A}_1 \mathbf{r} \\ \mathbf{G} &= \mathbf{A}_1^\top \mathbf{A}_1 - \mathbf{A}_1^\top \mathbf{B}_1 (\mathbf{B}_1^\top \mathbf{B}_1)^{-1} \mathbf{B}_1^\top \mathbf{A}_1 \end{aligned} \quad (7.51)$$

と書ける。上式は、記号は異なるものの式 (6.6) と式 (6.8) に等価である。

7.4 統一解法 2: EDPnPfr 法

本節では、DLT 法をベースとした部分問題 1 に対する効率的な手法 (Efficient DLT-based method, EDPnPfr 法) を提案する。部分問題 1 に対する手法であるから、部分問題 2 と root polishing については VPnPfr 法と同様である。VPnPfr 法との違いは、 $n \geq 6$ 点を対象としていること、および、平面と非平面に対するそれぞれの手法から構成されていることである。式 (7.22) を DLT 法に基づき定式化すると、

$$\begin{aligned} \min_{\mathbf{r}_1, \mathbf{r}_2} \quad & [\mathbf{r}_1^\top, \mathbf{r}_2^\top] \mathbf{M} \begin{bmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \end{bmatrix} \\ \text{s.t.} \quad & \|\mathbf{r}_1\|^2 + \|\mathbf{r}_2\|^2 = 1 \end{aligned} \quad (7.52)$$

である。式 (7.52) を解いた後、4.6 節にて述べた回転行列の補正を行う。次節より場合分けしたそれぞれの解法について説明するが、MATLAB 風の擬似コードを Algorithm 2 に記載する。

7.4.1 非平面

$n = 6$ 点の場合

点の数が $n = 6$ のとき、 $\text{rank}(\mathbf{M}) = 4$ である。そのため、未知パラータ $[\mathbf{r}_1^\top, \mathbf{r}_2^\top]$ は、 \mathbf{M} のゼロ空間ベクトル \mathbf{v}_1 と \mathbf{v}_2 の線形結合で表される。ゼロ空間ベクトルの係数を α とすると、

$$\begin{bmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \end{bmatrix} \propto \alpha \mathbf{v}_2 + \mathbf{v}_1, \quad \mathbf{v}_i = \begin{bmatrix} \mathbf{v}_{i1} \\ \mathbf{v}_{i2} \end{bmatrix} \quad (7.53)$$

である。この場合、式 (7.52) の目的関数は、 α の値に関係なく常にゼロである。

未知数 α を決定するために、 \mathbf{r}_1 と \mathbf{r}_2 に関する直交制約を用いる。これらの制約条件は次式で表される。

$$\mathbf{r}_1^\top \mathbf{r}_2 = (\alpha \mathbf{v}_{21} + \mathbf{v}_{11})^\top (\alpha \mathbf{v}_{22} + \mathbf{v}_{12}) = 0 \quad (7.54)$$

$$\|\mathbf{r}_1\|^2 - \|\mathbf{r}_2\|^2 = \|\alpha \mathbf{v}_{21} + \mathbf{v}_{11}\|^2 - \|\alpha \mathbf{v}_{22} + \mathbf{v}_{12}\|^2 = 0 \quad (7.55)$$

Algorithm 2 *Efficient DLT-based PnPfr Solver*

Input: $\mathbf{X} = [\mathbf{X}_1, \dots, \mathbf{X}_n], \mathbf{m} = \begin{bmatrix} u_1, \dots, u_n \\ v_1, \dots, v_n \end{bmatrix} \quad (n \geq 6)$

Output: $\mathbf{R}, \mathbf{t}, f, \mathbf{k}$

```

1: function [R, t, f, k] = EDPnPfr(X, m)
2:   [A, B, M] = calcABM(X, m) % 式 (7.20), 式 (7.23)
3:   if rank(cov(XT)) > 2 % 非平面
4:     [~, ~, V] = svd(M)
5:     if n == 6
6:       v1 = V(:, 6), v2 = V(:, 5)
7:       α = solveQuadratic(v1, v2) % 式 (7.56)
8:       [r1; r2] = αv2 + v1
9:     else
10:      [r1; r2] = V(:, 6)
11:    end
12:  else % 平面 (z = 0)
13:    M̂ = M([1 : 2, 4 : 5], [1 : 2, 4 : 5])
14:    [~, ~, V] = svd(M̂)
15:    [r̂1; r̂2] = V(:, 4)
16:    r132 = solveQuadratic(r̂1, r̂2) % 式 (7.60)
17:    r13 = ±√r132, r23 = -r̂1Tr̂2/r13
18:    [r1; r2] = [r̂1; r13; r̂2; r23]
19:  end
20:  R = recoverRotation(r1, r2) % 式 (7.37), 式 (4.14)
21:  [tx; ty] = -B \ A [r1; r2]
22:  [L, g] = calcLg(X, Y, R, tx, ty) % 式 (7.41)
23:  x = -L \ g
24:  tz = x(1)/x(2), f = 1/x(2), k = x(3 : 5)
25:  [R, t, f, k] = GaussNewton(X, Y, R, t, f, k) % 式 (7.48)
26: end

```

上式はいずれも α に関する 2 次式であるから、どちらかを閉形式で解けばよい。式 (7.54) を用いると、2 次式は

$$(\mathbf{v}_{21}^T \mathbf{v}_{22}) \alpha^2 + (\mathbf{v}_{21}^T \mathbf{v}_{12} + \mathbf{v}_{11}^T \mathbf{v}_{22}) \alpha + \mathbf{v}_{11}^T \mathbf{v}_{12} = 0 \quad (7.56)$$

と表される。結果として α について最大 2 つの実数解を得る。

 $n \geq 7$ 点の場合

非平面に分布する $n \geq 7$ 点が与えられた場合、ノイズのないデータに対しては $\text{rank}(\mathbf{M}) = 5$ 、ノイズのであるデータに対しては $\text{rank}(\mathbf{M}) = 6$ である。通常の DLT 法 (6.2.1 節) と同様に、式 (7.52) の解は最小固有値に対応する固有ベクトルとして得ら

れる。

7.4.2 平面

点分布が平面の場合、一般性を失わずに3次元点は平面 $z = 0$ 上に分布すると仮定できる。行列 A の第3列と第6列はゼロベクトルになるので、その2列を除いた 4×4 行列を \hat{M} とおく。これを用いると、式 (7.52) は以下のように書き換えられる。

$$\begin{aligned} \min_{\hat{\mathbf{r}}_1, \hat{\mathbf{r}}_2} \quad & [\hat{\mathbf{r}}_1^T, \hat{\mathbf{r}}_2^T] \hat{M} \begin{bmatrix} \hat{\mathbf{r}}_1 \\ \hat{\mathbf{r}}_2 \end{bmatrix} \\ \text{s.t.} \quad & \|\hat{\mathbf{r}}_1\|^2 + \|\hat{\mathbf{r}}_2\|^2 = 1 \end{aligned} \quad (7.57)$$

ここで、 $\hat{\mathbf{r}}_1 = [r_{11}, r_{12}]^T$ 、 $\hat{\mathbf{r}}_2 = [r_{21}, r_{22}]^T$ である。式 (7.57) と同様に、通常の DLT 法で解を求めることができる。

$\hat{\mathbf{r}}_1$ と $\hat{\mathbf{r}}_2$ が得られたので、 \mathbf{r}_1 と \mathbf{r}_2 の直交制約を利用して残りの未知数 r_{13} と r_{23} を算出する。式 (7.54)–(7.55) と同様に、次式を得る。

$$\hat{\mathbf{r}}_1^T \hat{\mathbf{r}}_2 + r_{13} r_{23} = 0 \quad (7.58)$$

$$\hat{\mathbf{r}}_1^T \hat{\mathbf{r}}_1 + r_{13}^2 - (\hat{\mathbf{r}}_2^T \hat{\mathbf{r}}_2 + r_{23}^2) = 0 \quad (7.59)$$

上式を用いて r_{13} の r_{23} どちらかを消去すると、もう一方についての4次式を得る。例えば r_{23} を消去すると、 r_{13} に関する4次式は

$$r_{13}^4 + (\hat{\mathbf{r}}_1^T \hat{\mathbf{r}}_1 - \hat{\mathbf{r}}_2^T \hat{\mathbf{r}}_2) r_{13}^2 - \hat{\mathbf{r}}_1^T \hat{\mathbf{r}}_2 = 0 \quad (7.60)$$

で表される。上式は r_{13}^2 に関する2次式とみなせるので、2次式の閉形式を用いて容易に解を算出できる。求めた解 $\pm \sqrt{r_{13}^2}$ を式 (7.58) に代入すると、対応する r_{23} の解は次式で表される。

$$r_{23} = -\frac{\hat{\mathbf{r}}_1^T \hat{\mathbf{r}}_2}{r_{13}} \quad (7.61)$$

7.5 実験

2つの提案手法 (VPnPfr 法と EDPnPfr 法) について、シミュレーションデータと実画像データを用いて評価実験を行い、従来手法と性能を比較した。シミュレーション実験では、点の数 n と画像ノイズを変化させながら PnP, PnPf, PnPfr 問題に対するパラメータの推定精度を調査した。また、実行時間の計測も行った。実画像実験では真値を得ることが困難なため、レンズ歪を取り除いた画像の定性的な比較を行った。

提案手法1と提案手法2は、各問題に対して次のように記載する。PnP 問題に対しては VPnP と EDPnP, PnPf 問題に対しては VPnPf と EDPnPf, PnPfr 問題に対しては VPnPfr と EDPnPfr とする。各問題において比較する手法は以下の通りである。従来手法の内容は6.2節と7.2節を参照のこと。

PnP 問題

DLT [24], EPnP+GN [46], OPnP [91], UPnP [36], VPnP, EDPnP

PnPf 問題

DLT [24], GPnPf+GN [93], EPnPfr [30], VPnPf, EDPnPf

PnPfr 問題

P5Pfr+NL [41] (P5Pfr+RANSAC+ 再投影誤差最小化), VPnPfr, EDPnPfr

提案手法はすべての MATLAB で実装した。従来手法の中で、UPnP 法は C++ の公開コードを MATLAB ラッパーを用いて呼び出し、P5Pfr 法は新規に MATLAB で実装した。その他の従来手法は公開されている MATLAB 実装を利用した。なお、すべての実験は Core i7-6700 を搭載する PC 上で実行した。

7.5.1 シミュレーション実験

3次元点は以下のように生成した。非平面の場合は、世界座標系における x - y - z が $[-2, 2] \times [-2, 2] \times [4, 8]$ の範囲内に、准平面の場合は世界座標系における $[-2, 2] \times [1, 2] \times [4, 8]$ の範囲内に、平面の場合はカメラ座標系における $[-2, 2] \times [-2, 2] \times [0, 0]$ の範囲内に、ランダムに生成した。広角レンズを想定したカメラの内部パラメータは焦点距離 $f = 500$ 、せん断係数 $\gamma = 0$ 、光学中心は画像中心とした。レンズ歪は、PnP, PnPf 問題において $\mathbf{k} = [0, 0, 0]^T$, PnPfr 問題において $\mathbf{k} = [-3/f^2, -0.5/f^4, -0.05/f^6]^T$ とした。カメラの外部パラメータを試行ごとにランダムに決定し、3次元点を画像平面へ射影した。パラメータの推定誤差は、式 (6.36) に示すように、PnP 問題と同様に回転行列については絶対誤差を、並進ベクトルについては相対誤差を算出した。

点数の変化に対するノイズ頑健性

本実験では点の数を $6 \leq n \leq 100$ と変化させた。画像ノイズは、平均ゼロ、標準偏差 $\sigma = 2$ ピクセルとする標準正規分布である。各点数 n について独立に 500 回試行し誤差を計測した。

図 7.1 に、回転行列 \mathbf{R} と焦点距離 f とレンズ歪の第 1 項 k_1 の誤差の中央値を示す。並進ベクトル \mathbf{t} の結果は回転行列と、レンズ歪の第 2 項 k_2 と第 3 項 k_3 の結果は第 1 項 k_1 と同じ傾向であるため記載は省略する。

提案手法 1 および 2 は、PnP 問題と PnPf 問題に対しては、いずれも従来手法の state-of-the-art と同精度な結果が得られた。提案手法 1 (VPnP, VPnPf, VPnPfr) は平面、准平面、非平面に対して一般性を持つ結果を示しており、部分問題に分割するアプローチが正しく動作することを実験的に証明している。一方、提案手法 2 (EDPnP, EDPnPf, EDPnPfr) は場合分けが必要なものの、准平面に対しても破綻することなく推定できており、シンプルな実装にも関わらずグレンナー基底を用いる提案手法 1 に遜色ない結果を志している。

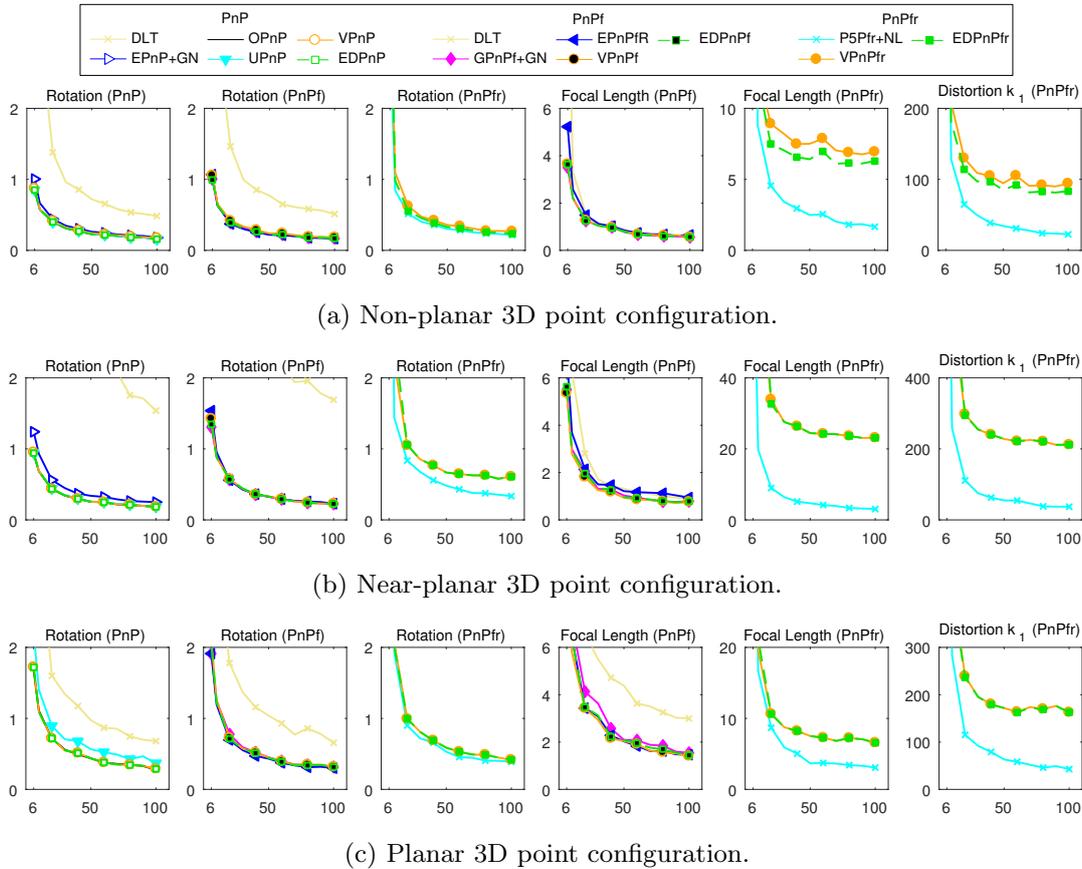


図 7.1: 画像ノイズを $\sigma = 2$ と固定, 点数を $6 \leq n \leq 100$ と変化させたときのノイズ頑健性

PnPfr 問題において, 提案手法は回転行列については P5Pfr+NL に匹敵するが, 内部パラメータの推定精度は及ばない結果となった. これは, 最小化する誤差関数の違い (提案手法: 代数的誤差, P5Pfr+NL: 再投影誤差) に起因すると思われる. この結果は PnPfr 問題における代数的誤差最小化の限界を示唆しているが, 回転行列の結果が肉薄していることを考慮すると, 内部パラメータについてもさらなる改善の可能性はある.

画像ノイズの変化に対する頑健性

本実験では, 点の数を $n = 20$ と固定し画像ノイズを $1 \leq \sigma \leq 5$ ピクセルと変化させた. 各ノイズレベル σ について独立に 500 回試行し誤差を計測し, 誤差の中央値を図 7.2 に示す.

PnP 問題と PnPf 問題では, 前節の実験と同様に, 提案手法 1 および 2 はいずれも state-of-the-art の従来手法と同等の精度を示した. 本実験でも准平面に対して良好な結果が得られており, 部分問題への分割による弊害が生じていないことを示している.

しかしながら, P5Pfr+NL と比較すると, VPnPfr と EDPnPfr の推定誤差はノイズの増加に対して急激に悪化する. 特に, 焦点距離とレンズ歪は画像ノイズに敏感である. 前

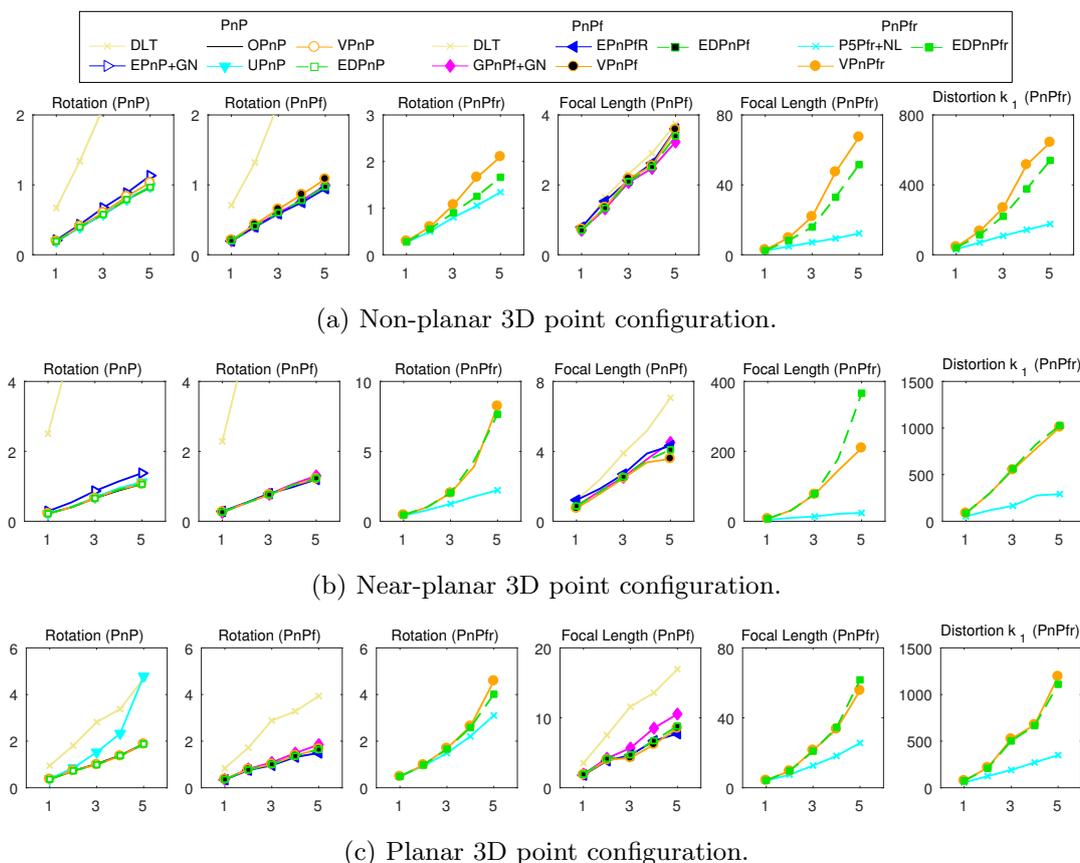


図 7.2: 点数を $n = 20$ と固定, 画像ノイズを $1 \leq \sigma \leq 5$ と変化させたときのノイズ頑健性

節でも述べたように, 本結果は代数的誤差の限界を示していることも考えられるが, 現段階では結論を下すことは難しい.

処理速度

画像ノイズを $\sigma = 2$ ピクセルと固定し, 点の数を $6 \leq n \leq 2000$ と変化させながら実行時間を計測した. 図 7.3 は, 各 n に対して独立に 500 回試行したときの平均値 (ミリ秒) に結果を示している. P5Pfr+NL の処理速度は 15 ミリ秒よりも長かったため記載していない.

まず, VPnPfr を除く提案手法 1 (VPnP 系) および 2 (EDnP 系) はいずれもほとんどの n に対して従来手法より高速である. 中でも, DLT 法に基づく EDPnP と EDPnPf は $n = 2000$ においても 3 ミリ秒未満であり, リアルタイム処理にも耐えうる速度を達成している. 一方, グレブナー基底を利用する VPnPfr はどの n に対しても 9 ミリ秒以上要しており, VPnP と VPnPf に比べて約 2 倍遅い. 最も大きな原因は, PnPfr 問題の root polishing において式 (7.47) が複雑なため数値微分 (numerical differentiation) を用いたことである. PnP 問題と PnPf 問題に対してはヤコビアンを書き下している. EDPnPfr も同様に数値微分を用いているが, 部分問題 1 の解法が高速なため 5 ミリ秒に収まっている

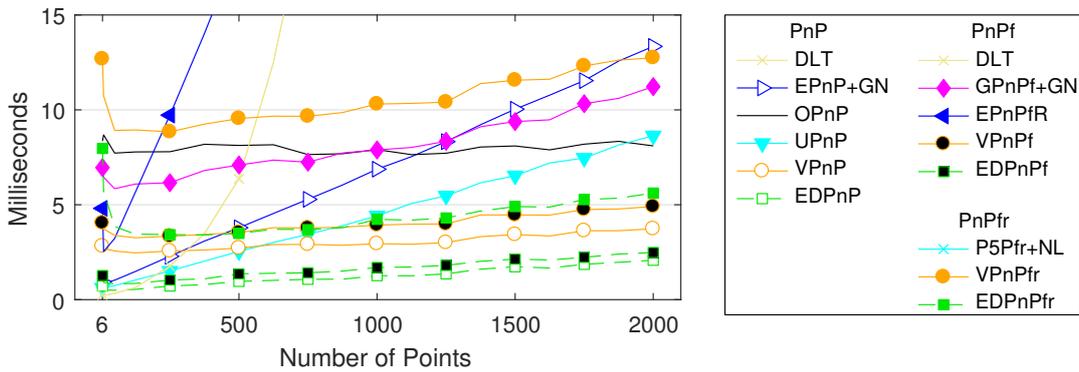


図 7.3: 画像ノイズを $\sigma = 2$ と固定, 点数を $6 \leq n \leq 2000$ と変化させたときの平均実行時間

表 7.2: 実画像に対する実験結果. 画像解像度は 1024×1024 . 水平画角 118° の広角レンズを使用.

Parameter	Spec sheet	P5Pfr+NL [41]	VPnPfr	EDPnPfr
HFOV	118°	111.58°	111.60°	111.60°
焦点距離 f	307.64	348.11	347.90	347.90
レンズ歪 k_1	n.a.	-3.43×10^{-6}	-3.41×10^{-6}	-3.41×10^{-6}
k_2	n.a.	-2.05×10^{-12}	-2.57×10^{-12}	-2.57×10^{-12}
k_3	n.a.	-2.57×10^{-18}	5.01×10^{-19}	5.01×10^{-19}

る. 式 (7.47) のヤコビアンを導出したり, 自動微分 (automatic differentiation) を用いたりすることも可能だが, 実行時間はおそらく 7 ミリ秒程度と予想される. そのため, グレブナー基底を用いた PnPfr 問題の解法には高速化の課題が残されている.

7.5.2 実画像実験

PnPfr 問題の解法である P5Pfr+NL, VPnPfr, EDPnPfr を実画像にて評価した. ズーム倍率を変更可能な TAMRON 製レンズ 12VM412ASIR を, iDS 製カメラ UI-3370CP-C-HQ に装着して画像を撮影した. 画像解像度は 1024×1024 ピクセルである. レンズのスペック表によると, 水平画角 (horizontal field of view, HFOV) は最小 58° , 最大 118° であり, 最大画角における焦点距離は 308.64 ピクセルに相当する. 真値の測定は困難なため, 手法間における推定値の比較と, 歪みパラメータを用いて歪補正した画像を定性的に比較する.

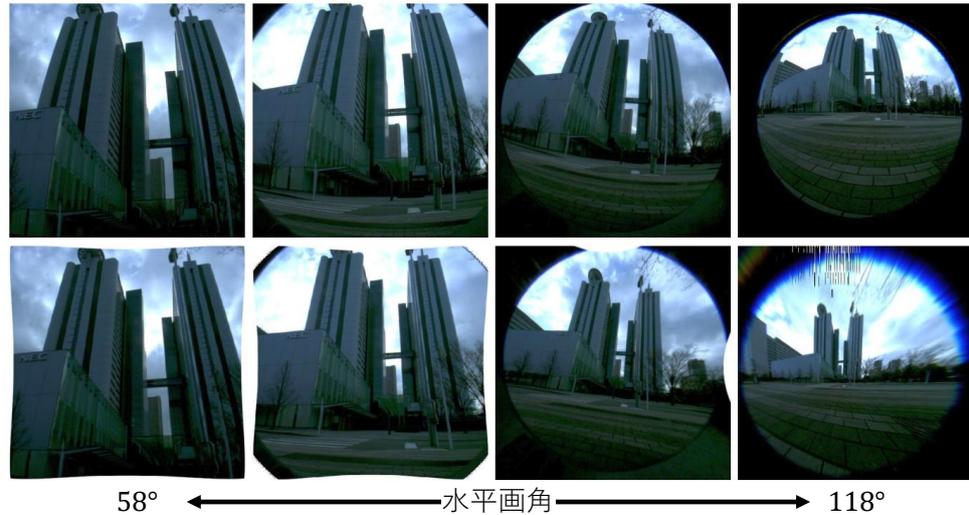


図 7.4: 水平画角を変化させた画像に対する提案手法 VPnPfr による歪み除去. 上段: 撮影画像, 下段: 補正画像.

チェッカーボードを含む画像を撮影し, 格子点を平面上の 3 次元点として 3 つの手法を適用した. 表 7.2 に原画像と補正後の画像, および推定した内部パラメータの値を示す. 画像の歪みは正確に補正され, チェッカーボードだけでなく本や机のエッジも正しく直線に補正されていることが確認できる. 手法間での推定値はほぼ同じで, いずれもカタログ値に近い値が推定できている.

加えて, 同レンズの水平画角を 58 度から 118 度まで変化させて撮影した画像に対して, 提案手法 VPnPfr により歪み除去を行った結果を図 7.4 に示す. 各ズーム値でチェッカーボードを撮影して歪みパラメータを推定した. 本結果は, 例えば VSLAM の実行中にカメラのズーム値が変動しても, 提案手法ではカメラパラメータが精度良く推定可能なことを示している.

7.6 考察

シミュレーション実験 (7.5.1 節) の結果より, 部分問題に分割するという提案手法のアイデアは適切に動作し, 3 つの問題 (PnP, PnPf, PnPfr) を同一のアプローチで解けることが実証された. また, PnPfr 問題においては再投影誤差最小化を用いる手法よりも低精度であったが, 実際には誤差関数の違いによる影響は少ないことが実画像実験 (7.5.2) にて確認された. 処理速度 (7.5.1 節) を考慮に入れると, 提案手法は再投影誤差最小化よりも実用的であると結論付けられる.

7.7 第7章のまとめ

本章では、まず代数的誤差に基づく PnPfr 問題の定式化を行い、PnP 問題と PnPf 問題が包含されることを示した。次に、PnPf 問題と PnPfr 問題に対する従来手法を簡単に説明した。そして、PnPfr 問題を2つの部分問題に分割することで、3つの問題を同一アプローチで解く手法を2つ提案した。1つ目の手法はグレブナー基底を用いて、2つ目の手法は DLT 法に基づいて、第1の部分問題を解く。シミュレーション実験の結果により、提案手法のアプローチは問題なく動作し、3問題が解けることを実証した。また、PnPfr 問題のシミュレーション実験では提案手法の精度は再投影誤差最小化に劣るものの、実環境ではほぼ同性能を高速に達成するため問題にはならないことを示した。

本章の実験については、第5章や第6章と同様に、入力である3次元点のノイズについて検討が必要である。提案手法独自の懸念としては、第1の部分問題がカメラ座標系の z 軸方向のみから立式されていることである。3次元点の推定に例えばステレオ画像を用いると、推定誤差は非等方性であり奥行き方向の誤差が比較的大きいことが知られている。同様に、第1の部分問題は奥行き方向の推定のみを行うため、第6章で提案した PnP 問題への解法よりも入力のノイズに敏感だと推測される。これらを検証することで、PnPfr 問題において代数的誤差に基づきながらも、再投影誤差最小化に匹敵する性能を持つ手法を提案できる可能性がある。また、PnPf 問題と PnPfr 問題における理論的に推定が不可能となるカメラと点群の位置関係、いわゆる退化条件 (Degeneracy condition) の解析は十分に明らかにされていない。安定的に推定を行うためには、退化条件の数学的に詳細な解析は今後不可欠である。

本章に関連する公表論文は以下の通りである。

Gaku Nakano, A Versatile Approach for Solving PnP, PnPf, and PnPfr Problems, the 14th European Conference on Computer Vision (ECCV2016), 2016.

Gaku Nakano, Efficient DLT-based Method for Solving PnP, PnPf, and PnPfr Problems, IEICE Trans Info & Sys, (under review).

本章にて提案した手法のコードは以下のウェブサイトにて公開している。

https://github.com/g9nkn/pnpfr_problem

7.8 付録：グレブナー基底自動計算ツール

7.3.2 節にて説明した、第1の部分問題をグレブナー基底を利用して解くための MATLAB コードを掲載する。

Kukelova ら [38] のツール

```

% register the generator
setpaths;

% coefficient matrix (6x6 symmetric)
M = gbs_Matrix('M%d%d', 6, 6);
M = M - tril(M) + triu(M).';

% unknowns (1st and 2nd row of R)
syms a b c d e
r1 = [a; b; c];
r2 = [d; e; 1]; %linear constraint, R(3,3)=1

% build polynomial equations
A = M(1:3,1:3)*r1 + M(1:3,4:6)*r2;
B = M(1:3,4:6).'*r1 + M(4:6,4:6)*r2;
S1 = [0 -c b; c 0 -a; -b a 0]; % [r1]_x
S2 = [0 -1 e; 1 0 -d; -e d 0]; % [r2]_x
eq1 = S1*A + S2*B;
eq2 = r2.*A - r1.*B;
ceq1 = r1.*r1 - r2.*r2;
ceq2 = r1.*r2;
eqs = [ceq1, ceq2, eq1(:).', eq2];

% collect known & unknown variables
unknown = {'a','b', 'c', 'd','e'};
vars = symvar(M);
known = {};
for var = vars
    known = [known {char(var)}];
end

% call gbsolver
[res, export] = gbs_CreateCode(mfilename, eqs, known, unknown);

```

Larsson ら [42] のツール

```
opts = default_options();
solv = generate_solver('vnpnfr', @problem_vnpnfr, opts);

function [ eqs, data0, eqs_data ] = problem_vnpnfr(data0)

if nargin < 1 || isempty(data0)
M = randi(30097,6,6);
M = M+M';
data0 = M(:);
end

M = reshape(data0,6,6);
xx = create_vars(5);

r1 = [xx(1); xx(2); xx(3)];
r2 = [xx(4); xx(5); 1]; %linear constraint, R(2,3)=1

% build polynomial equations
A = M(1:3,1:3)*r1 + M(1:3,4:6)*r2;
B = M(1:3,4:6)'*r1 + M(4:6,4:6)*r2;
S1 = [ 0, -r1(3), r1(2);
r1(3), 0, -r1(1);
-r1(2), r1(1), 0]; % [r1]_x
S2 = [ 0, -r2(3), r2(2);
r2(3), 0, -r2(1);
-r2(2), r2(1), 0]; % [r2]_x
eq1 = S1*A + S2*B;
eq2 = r2'*A - r1'*B;
ceq1 = r1'*r1 - r2'*r2;
ceq2 = r1'*r2;
eqs = [eq1(:); eq2; ceq1; ceq2];

if nargin == 3
xx = create_vars(5+36);
eqs_data = problem_vnpnfr(xx(6:end));
end

end
```


第 8 章

結論

本論文では PnP 問題とその派生問題である PnPf 問題および PnPfr 問題について議論した。まず、第 2 章にて PnP 問題の歴史を振り返り、4 つの課題（安定性、高速性、一般性、拡張性）に対してどのような手法が提案されてきたかを説明した。次に第 3 章で PnP 問題に関連が深い研究を紹介し、第 4 章にて本論文で用いるカメラの射影幾何学について説明した。そして、第 5–7 章にかけて、PnP 問題とその派生問題について新たな手法を提案した。

本論文で提案した手法は以下の通りである。

第 5 章にて、P3P 問題に対し回転行列を射影的奥行き線形結合で表す直接解法を提案した。Banno の手法と数学的には等価であるが、計算をより効率化させた手法である。シミュレーション実験により、高速性と安定性のバランスが取れた手法であること実証した。また、カメラと 3 点を作る三角形が互いに正対する配置では、従来手法の多くにアルゴリズムが破綻する退化条件が内在することを示した。

第 6 章では、PnP 問題に対してグレブナー基底を用いて局所解をすべて列挙する、大域的に最適な解法を提案した。まず、ラグランジュ乗数を用いずに回転行列の制約を表現する新たな最適性条件を導出した。導いた最適性条件を多項式の連立方程式として解くことで、平面と非平面を問わずすべての局所解が列挙できる。これにより、異なる回転行列の表現を用いても解空間が同一であることを示し、中でも Cayley 変換が最も効率的な表現であることを指摘した。シミュレーション実験により、従来不安定とみなされてきた Cayley 変換を用いたソルバーは、従来手法と同等の安定性と一般性を持ちながら、点の数を増やしたときの拡張性と高速性に優れた手法であることを示した。

第 7 章では、PnPfr 問題を 2 つの部分問題で分割することで、3 問題を同一アプローチで解ける統一的な解法を 2 つ提案した。第 1 の部分問題は第 6 章で議論した PnP 問題と類似しており、回転行列と第 3 成分を除く並進ベクトルとで構成される。そのため、PnP、PnPf、PnPfr 問題に共通なパラメータから構成されている。第 1 の部分問題で得られる解を既知とおくと、第 2 の部分問題は残りのパラメータに対して線形最小二乗法で表される。最初の提案手法は第 1 の部分問題をグレブナー基底を用いて解き、2 つ目の提案手法は DLT 法に基づいた線形近似により第 1 の部分問題を解く。シミュレーション実

験の結果により、部分問題に分割することで 3 問題を共通のアプローチで解けることを示した。グレブナー基底に基づく提案手法と DLT 法に基づく提案手法はいずれも同精度であるが、実行速度の面では DLT 法に基づく第 2 の手法が有利である。一方で、グレブナー基底に基づく第 1 の手法は平面と非平面を区別しない一般性において優れている。また、ノイズ頑健性については再投影誤差最小化を用いる従来手法に劣るが、対応点が高精度に取得できれば実用上問題にはならないことを実画像実験にて示した。

本論文で提案した手法および PnP 問題に関するいくつかの課題について述べる。まず、第 5 章で示したように、完成されていると思われた P3P 問題において従来手法のアルゴリズムに退化条件が存在することが明らかになった。これまで解の個数やカメラ配置に関する解析 [20] は行われてきたが、アルゴリズムごとの詳細な解析は行われていない。単に効率的な 4 次式への帰着を競うだけでなく、真に安定な手法を評価する包括的な研究が必要である。同様に、PnPf 問題と PnPfr 問題についても退化条件について理論的な解析が不可欠である。

次に、本論文の提案手法を含め代数的誤差を用いた場合に解が持つ偏差についての解析は不十分である。PnP 問題に対しては解の偏差を除去する発見的な反復解法 [16] が提案されているものの、グレブナー基底を用いるような大域的に最適手法の研究はいまだ提案されていない。また、PnPf 問題や PnPfr 問題については解の偏差について議論されていない。特に、第 7 章で述べたように、PnPfr 問題における代数的誤差最小化はまだまだ改善の余地がある。誤差伝搬の高次解析 [32] は、再投影誤差に匹敵する手法の導出に寄与する可能性がある。

上記課題に関連して、PnP 問題とその派生問題は内部パラメータと 3 次元点にノイズが無いと仮定している。しかし、現実にはどちらにも計測誤差が乗るが、どのような誤差であるかは手法により異なる。格子パターンを用いる焦点距離推定の計測誤差 [33] やステレオ画像による 3 次元点の誤差分布 [31] など限定した対象については研究が行われているが、カメラキャリブレーションや SfM において誤差分布を取り扱った包括的な研究はされていない。PnP 問題のアプリケーションである VSLAM や AR において、誤差分布を特徴点ごとに格納すれば、より安定的なシステムが構築できると思われる。

謝辞

本論文をまとめるにあたり、数多くの方々からご指導とご助力を賜りました。お世話になった皆様に、この場をお借りして感謝の意を申し上げます。

はじめに、主査の筑波大学 北原格教授には、社会人として博士課程へ進学することを強く後押ししていただいただけでなく、本論文の執筆のために専門的な内容に限らず公私に渡り様々な助言をいただきました。私生活、会社生活、博士課程とのバランスが取れたのも、ひとえに北原先生のおかげです。心より感謝申し上げます。

学位審査をご快諾いただきました副査の先生方からは、本論文をまとめる上で数多くの有益な助言をいただきました。筑波大学 坪内孝司教授には、シミュレーション実験におけるノイズモデルの正当性について助言をいただきました。また、発表論文をただまとめるだけでなく、読み応えのある内容にすべくご指摘をいただきました。筑波大学 亀田能成教授には、代数学的誤差の最適性条件における解釈について助言をいただき、導出した私自身が気づいていなかった点について指摘していただきました。国立情報学研究所 杉本晃宏教授には、代数的な導出に基づく提案手法に対して幾何的な解釈を与えるようご指摘をいただきました。産業技術総合研究所 櫻田健主任研究員には、VSLAM の専門家として実用面からの助言をいただきました。多方面からの助言をいただき論文の完成度を高めることができたこと、厚く御礼申し上げます。

そして、筑波大学 大田友一名誉教授からは、研究者としての心構えを学びました。学士-修士課程にてご指導いただいたことは、今になってようやく理解できるようになりました。研究の枝葉末節に囚われて思考が偏ってしまったときは、先生の言葉を思い出しています。

また、私が所属している日本電気株式会社中央研究所の皆様にも、多くの支えをいただきました。幡生敦史さんには、同じ部に所属しているときに数学的な雑談に付き合ってくださいました。あのときの雑談がなければ本論文の一部は誕生しませんでした。山田昭雄執行役員には、情報メディアプロセッシング研究所所長であられた当時 2014 年にカーネギーメロン大学への留学のチャンスをいただきました。海外留学は私が企業研究者としてのあり方を見つめ直し、博士号を意識する大きな転機になりました。現 Yahoo! Japan 研究所 杉山昭彦特任研究員には、NEC 在籍時代に論文の書き方からアカデミア活動の重要性まで幅広く学ばせていただきました。

最後に、私の論文発表を我が事のように喜び励ましてくれる母京子、そして精神的にも

時間的にも支えてくれる妻寛子に深く感謝します。新型コロナウイルスの蔓延による突然のリモートワークにも関わらず、家事と子育てを一手に引き受けて論文執筆のための時間を割いてくれました。二人の娘、桐子と桃子には遊びたい盛りに寂しい思いをさせてしまいました。家族の支援がなければ、この論文を完成させることは不可能でした。本当にありがとう。

中野 学

参考文献

- [1] Sameer Agarwal, Keir Mierle, and Others. Ceres solver. <http://ceres-solver.org>.
- [2] Alexandre Alahi, Raphael Ortiz, and Pierre Vandergheynst. Freak: Fast retina keypoint. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 510–517. Ieee, 2012.
- [3] Cenek Albl, Zuzana Kukelova, and Tomas Pajdla. R6p-rolling shutter absolute camera pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2292–2300, 2015.
- [4] Cenek Albl, Zuzana Kukelova, and Tomas Pajdla. Rolling shutter absolute pose problem with known vertical direction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3355–3363, 2016.
- [5] Adnan Ansar and Konstantinos Daniilidis. Linear pose estimation from points or lines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):578–589, 2003.
- [6] Erik Ask, Yubin Kuang, and Kalle Astrom. Exploiting p-fold symmetries for faster polynomial equation solving. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 3232–3235. IEEE, 2012.
- [7] Atsuhiko Banno. A p3p problem solver representing all parameters as a linear combination. *Image and Vision Computing*, 70:55–62, 2018.
- [8] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European conference on computer vision*, pages 404–417. Springer, 2006.
- [9] Duane Brown. Close-range camera calibration. *Photogramm. Eng*, 37(8):855–866, 1971.
- [10] Martin Bujnak, Zuzana Kukelova, and Tomas Pajdla. A general solution to the p4p problem for camera with unknown focal length. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.
- [11] Martin Bujnak, Zuzana Kukelova, and Tomas Pajdla. New efficient solution to the absolute pose problem for camera with unknown focal length and radial

- distortion. In *Asian Conference on Computer Vision*, pages 11–24. Springer, 2010.
- [12] Martin Byrod, Zuzana Kukelova, Klas Josephson, Tomas Pajdla, and Kalle Astrom. Fast and robust numerical solutions to minimal problems for cameras with radial distortion. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.
- [13] David Claus and Andrew W Fitzgibbon. A rational function lens distortion model for general cameras. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 213–219. IEEE, 2005.
- [14] David A Cox, John Little, and Donal O'shea. *Using algebraic geometry*, volume 185. Springer Science & Business Media, 2006.
- [15] David W Eggert, Adele Lorusso, and Robert B Fisher. Estimating 3-d rigid body transformations: a comparison of four major algorithms. *Machine vision and applications*, 9(5-6):272–290, 1997.
- [16] Luis Ferraz Colomina, Xavier Binefa, and Francesc Moreno-Noguer. Leveraging feature uncertainty in the pnp problem. In *Proceedings of the BMVC 2014 British Machine Vision Conference*, pages 1–13, 2014.
- [17] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
- [18] Andrew W Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001.
- [19] Friedrich Fraundorfer, Petri Tanskanen, and Marc Pollefeys. A minimal case solution to the calibrated relative pose problem for the case of two known orientation angles. In *European Conference on Computer Vision*, pages 269–282. Springer, 2010.
- [20] Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang, and Hang-Fei Cheng. Complete solution classification for the perspective-three-point problem. *IEEE transactions on pattern analysis and machine intelligence*, 25(8):930–943, 2003.
- [21] Marcel Geppert, Viktor Larsson, Pablo Speciale, Johannes L. Schönberger, and Marc Pollefeys. Privacy preserving structure-from-motion. In *ECCV, 2020*.
- [22] Johann August Grunert. Das pothenotische problem in erweiterter gestalt nebst bber seine anwendungen in der geodasie. *Grunerts Archiv fur Mathematik und Physik*, pages 238–248, 1841.
- [23] Bert M Haralick, Chung-Nan Lee, Karsten Ottenberg, and Michael Nölle. Review

- and analysis of solutions of the three point perspective pose estimation problem. *International journal of computer vision*, 13(3):331–356, 1994.
- [24] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.
- [25] Janne Heikkila. Geometric camera calibration using circular control points. *IEEE Transactions on pattern analysis and machine intelligence*, 22(10):1066–1077, 2000.
- [26] Joel A Hesch and Stergios I Roumeliotis. A direct least-squares (dls) method for pnp. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 383–390. IEEE, 2011.
- [27] Hatem Hmam and Jijoong Kim. Optimal non-iterative pose estimation via convex relaxation. *Image and Vision Computing*, 28(11):1515–1523, 2010.
- [28] Fangyuan Jiang, Yubin Kuang, Jan Erik Solem, and Kalle Åström. A minimal solution to relative pose with unknown focal length and radial distortion. In *Asian Conference on Computer Vision*, pages 443–456. Springer, 2014.
- [29] Klas Josephson and Martin Byrod. Pose estimation with radial distortion and unknown focal length. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2419–2426. IEEE, 2009.
- [30] Ekaterina Kanaeva, Lev Gurevich, and Alexander Vakhitov. Camera pose and focal length estimation using regularized distance constraints. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 162.1–162.12. BMVA Press, September 2015.
- [31] Kenichi Kanatani. *Statistical optimization for geometric computation: theory and practice*. Elsevier, 1996.
- [32] Kenichi Kanatani. Statistical optimization for geometric estimation: minimization vs. non-minimization. In *2014 22nd International Conference on Pattern Recognition*, pages 1–8. IEEE, 2014.
- [33] Kenichi Kanatani and Tamotsu Maruyama. Optimal focal length calibration system. In *Proceedings of 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'93)*, volume 3, pages 1816–1821. IEEE, 1993.
- [34] Hirokazu Kato and Mark Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99)*, pages 85–94. IEEE, 1999.
- [35] Tong Ke and Stergios I Roumeliotis. An efficient algebraic solution to the perspective-three-point problem. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7225–7233, 2017.
- [36] Laurent Kneip, Hongdong Li, and Yongduek Seo. Upnp: An optimal o (n)

- solution to the absolute pose problem with universal applicability. In *Computer Vision–ECCV 2014*, pages 127–142. Springer, 2014.
- [37] Laurent Kneip, Davide Scaramuzza, and Roland Siegwart. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2969–2976. IEEE, 2011.
- [38] Zuzana Kukelova, Martin Bujnak, and Tomas Pajdla. Automatic generator of minimal problem solvers. In *European Conference on Computer Vision*, pages 302–315. Springer, 2008.
- [39] Zuzana Kukelova, Martin Bujnak, and Tomas Pajdla. Closed-form solutions to minimal absolute pose problems with known vertical direction. In *Asian Conference on Computer Vision*, pages 216–229. Springer, 2010.
- [40] Zuzana Kukelova, Martin Bujnak, and Tomas Pajdla. Polynomial eigenvalue solutions to minimal problems in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7):1381–1393, 2011.
- [41] Zuzana Kukelova, Martin Bujnak, and Tomas Pajdla. Real-time solution to the absolute pose problem with unknown radial distortion and focal length. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2816–2823, 2013.
- [42] Viktor Larsson, Kalle Astrom, and Magnus Oskarsson. Efficient solvers for minimal problems by syzygy-based reduction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 820–829, 2017.
- [43] Viktor Larsson, Zuzana Kukelova, and Yinqiang Zheng. Making minimal solvers for absolute pose estimation compact and robust. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2316–2324, 2017.
- [44] Viktor Larsson, Torsten Sattler, Zuzana Kukelova, and Marc Pollefeys. Revisiting radial distortion absolute pose. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1062–1071, 2019.
- [45] Louis Lecrosnier, Rémi Boutteau, Pascal Vasseur, Xavier Savatier, and Friedrich Fraundorfer. Camera pose estimation based on pnl with a known vertical direction. *IEEE Robotics and Automation Letters*, 4(4):3852–3859, 2019.
- [46] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate $O(n)$ solution to the pnp problem. *International journal of computer vision*, 81(2):155–166, 2009.
- [47] Hongdong Li. A simple solution to the six-point two-view focal-length problem. In *European Conference on Computer Vision*, pages 200–213. Springer, 2006.
- [48] Shiqi Li, Chi Xu, and Ming Xie. A robust $O(n)$ solution to the perspective-n-point problem. *IEEE transactions on pattern analysis and machine intelligence*,

-
- 34(7):1444–1450, 2012.
- [49] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.
- [50] Chien-Ping Lu, Gregory D Hager, and Eric Mjolsness. Fast and globally convergent pose estimation from video images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(6):610–622, 2000.
- [51] Andreas Masselli and Andreas Zell. A new geometric approach for faster solving the perspective-three-point problem. In *2014 22nd International Conference on Pattern Recognition*, pages 2119–2124. IEEE, 2014.
- [52] Francesc Moreno-Noguer, Vincent Lepetit, and Pascal Fua. Accurate non-iterative $\mathcal{O}(n)$ solution to the pnp problem. In *2007 IEEE 11th International Conference on Computer Vision*, pages 2252–2259. IEEE, 2007.
- [53] Raúl Mur-Artal, J. M. M. Montiel, and Juan D. Tardós. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.
- [54] Gaku Nakano. Globally optimal dls method for pnp problem with cayley parameterization. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 78.1–78.11. BMVA Press, September 2015.
- [55] Gaku Nakano. A versatile approach for solving pnp, pnpf, and pnpfr problems. In *The 14th European Conference on Computer Vision (ECCV2016)*, pages 338–352. Springer, 2016.
- [56] Gaku Nakano. A simple direct solution to the perspective-three-point problem. In *Proceedings of the 30th British Machine Vision Conference (BMVC2019)*. BMVA Press, 2019.
- [57] Gaku Nakano and Jun Takada. A robust least squares solution to the relative pose problem on calibrated cameras with two known orientation angles. In *VISAPP (2)*, pages 147–154, 2013.
- [58] David Nistér. An efficient solution to the five-point relative pose problem. *IEEE transactions on pattern analysis and machine intelligence*, 26(6):756–770, 2004.
- [59] Mikael Persson and Klas Nordberg. Lambda twist: an accurate fast robust perspective three point (p3p) solver. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 318–332, 2018.
- [60] Francesco Pittaluga, Sanjeev J Koppal, Sing Bing Kang, and Sudipta N Sinha. Revealing scenes by inverting structure from motion reconstructions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 145–154, 2019.
- [61] Bronislav Pribyl, Pavel Zemčík, and Martin Čadik. Camera pose estimation from

- lines using plücker coordinates. In Mark W. Jones Xianghua Xie and Gary K. L. Tam, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 45.1–45.12. BMVA Press, September 2015.
- [62] Long Quan and Zhongdan Lan. Linear n-point camera pose determination. *IEEE Transactions on pattern analysis and machine intelligence*, 21(8):774–780, 1999.
- [63] Jürgen Richter-Gebert. *Perspectives on projective geometry: A guided tour through real and complex geometry*. Springer-Verlag Berlin Heidelberg, 2011.
- [64] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571. IEEE, 2011.
- [65] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *CVPR*, 2020.
- [66] Olivier Saurer, Mare Pollefeys, and Gim Hee Lee. A minimal solution to the rolling shutter pose estimation problem. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1328–1334. IEEE, 2015.
- [67] Jochen Schmidt and Heinrich Niemann. Using quaternions for parametrizing 3-d rotations in unconstrained nonlinear optimization. In Thomas Ertl, Bernd Girod, Heinrich Niemann, and Hans-Peter Seidel, editors, *6th International Fall Workshop on Vision, Modeling, and Visualization, VMV 2001, Stuttgart, Germany, November 21-23, 2001*, pages 399–406. Aka GmbH, 2001.
- [68] Gerald Schweighofer and Axel Pinz. Robust pose estimation from a planar target. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(12):2024–2030, 2006.
- [69] Gerald Schweighofer and Axel Pinz. Globally optimal $O(n)$ solution to the pnp problem for general camera models. In *BMVC*, pages 1–10, 2008.
- [70] Mikiya Shibuya, Shinya Sumikura, and Ken Sakurada. Privacy preserving visual SLAM. In *European Conference on Computer Vision (ECCV)*, 2020.
- [71] Ken Shoemake. Animating rotation with quaternion curves. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pages 245–254, 1985.
- [72] ADN Smith. The explicit solution of the single picture resection problem, with a least squares adjustment to redundant control. *The Photogrammetric Record*, 5(26):113–122, 1965.
- [73] Henrik Stewenius, Christopher Engels, and David Nistér. Recent developments on direct relative orientation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 60(4):284–294, 2006.
- [74] Henrik Stewenius, David Nistér, Fredrik Kahl, and Frederik Schaffalitzky. A

-
- minimal solution for relative pose with unknown focal length. *Image and Vision Computing*, 26(7):871–877, 2008.
- [75] Shinya Sumikura, Mikiya Shibuya, and Ken Sakurada. Openvslam: a versatile visual slam framework. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 2292–2295, 2019.
- [76] Chris Sweeney, John Flynn, Benjamin Nuernberger, Matthew Turk, and Tobias Höllerer. Efficient computation of absolute pose for gravity-aware augmented reality. In *2015 IEEE International Symposium on Mixed and Augmented Reality*, pages 19–24. IEEE, 2015.
- [77] George Terzakis, Phil Culverhouse, Guido Bugmann, Sanjay Sharma, and Robert Sutton. A recipe on the parameterization of rotation matrices for non-linear optimization using quaternions. Technical report, Tech. Rep. 004, School of Marine Science and Engineering. Plymouth University, 2012.
- [78] EH Thompson. Space resection: Failure cases. *The Photogrammetric Record*, 5(27):201–207, 1966.
- [79] Roger Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal on Robotics and Automation*, 3(4):323–344, 1987.
- [80] Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 13(4):376–380, 1991.
- [81] Steffen Urban, Jens Leitloff, and Stefan Hinz. Mlpnp - a real-time maximum likelihood solution to the perspective-n-point problem. In *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, volume 3, pages 131–138, 2016.
- [82] Carl Toft Viktor Larsson, Johan Fredriksson and Fredrik Kahl. Outlier rejection for absolute pose estimation with known orientation. In Edwin R. Hancock Richard C. Wilson and William A. P. Smith, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 45.1–45.12. BMVA Press, September 2016.
- [83] Ping Wang, Guili Xu, Yuehua Cheng, and Qida Yu. Camera pose estimation from lines: a fast, robust and general method. *Machine Vision and Applications*, 30(4):603–614, 2019.
- [84] Qianqian Wang, Xiaowei Zhou, Bharath Hariharan, and Noah Snavely. Learning feature descriptors using camera pose supervision. In *Proc. European Conference on Computer Vision (ECCV)*, 2020.
- [85] Mark D Wheeler and Katsushi Ikeuchi. *Iterative estimation of rotation and translation using the quaternion*. Carnegie-Mellon University. Department of

- Computer Science, 1995.
- [86] Changchang Wu. P3. 5p: Pose estimation with unknown focal length. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2440–2448, 2015.
- [87] C. Xu, L. Zhang, L. Cheng, and R. Koch. Pose estimation from line correspondences: A complete analysis and a series of solutions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.
- [88] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. Lift: Learned invariant feature transform. In *European Conference on Computer Vision*, pages 467–483. Springer, 2016.
- [89] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22(11):1330–1334, 2000.
- [90] Yinqiang Zheng and Laurent Kneip. A direct least-squares solution to the pnp problem with unknown focal length. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1790–1798, 2016.
- [91] Yinqiang Zheng, Yubin Kuang, Shigeki Sugimoto, Kalle Astrom, and Masatoshi Okutomi. Revisiting the pnp problem: A fast, general and optimal solution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2344–2351, 2013.
- [92] Yinqiang Zheng, Shigeki Sugimoto, and Masatoshi Okutomi. Aspnp: An accurate and scalable solution to the perspective-n-point problem. *IEICE TRANSACTIONS on Information and Systems*, 96(7):1525–1535, 2013.
- [93] Yinqiang Zheng, Shigeki Sugimoto, Imari Sato, and Masatoshi Okutomi. A general and simple method for camera pose and focal length determination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 430–437, 2014.
- [94] 中野学, 田治米純二, and 野村俊之. 一般カメラモデルの PnP 問題に対するグレブナー基底を用いた統一的解法. *画像の認識・理解シンポジウム (MIRU2011)*, pages 845–851, 2011.
- [95] 中野学, 田治米純二, and 野村俊之. 全停留点の直接計算に基づく一般カメラモデルの PnP 問題に対する統一的解法. *電子情報通信学会論文誌 D*, 95(8):1565–1572, 2012.
- [96] 日比孝之. *グレブナー基底*. 朝倉書店, 2003.
- [97] 金谷健一. *3次元回転：パラメータ計算とリー代数による最適化*. 共立出版, 2019.

本研究に関する公表済み論文

査読付き論文誌

1. 中野学, 田治米純二, 野村俊之, 全停留点の直接計算に基づく一般カメラモデルの PnP 問題に対する統一的解法, 電子情報通信学会論文誌 D, vol.95, no.8, pp.1565–1572, 2012.
2. Gaku Nakano, Efficient DLT-based Method for Solving PnP, PnPf, and PnPfr Problems, IEICE Transactions on Information & Systems, (under review).

査読付き国際会議

1. Gaku Nakano, A Simple Direct Solution to the Perspective-three-Point Problem, In Proceeding of the 30th British Machine Vision Conference (BMVC2019), BMVA Press, 2019.
2. Gaku Nakano, A Versatile Approach for Solving PnP, PnPf, and PnPfr Problems, In Proceeding of the 14th European Conference on Computer Vision (ECCV2016), pp.338–352, Springer, 2016.
3. Gaku Nakano, Globally Optimal DLS Method for PnP Problem with Cayley parameterization, In Proceeding of the 26th British Machine Vision Conference (BMVC2015), pp.78.1–78.11, BMVA Press, 2015.

その他の発表論文

1. 中野学, 田治米純二, 野村俊之, 一般カメラモデルの PnP 問題に対するグレブナー基底を用いた統一的解法, 画像の認識・理解シンポジウム (MIRU2011), pp.845–851.