

NAIST-IS-MT1351040

修士論文

特徴点の明示的な対応付けを必要としない カメラ位置姿勢推定

黒川 陽平

2015年3月11日

奈良先端科学技術大学院大学
情報科学研究科 情報科学専攻

本論文は奈良先端科学技術大学院大学情報科学研究科に
修士(工学) 授与の要件として提出した修士論文である。

黒川 陽平

審査委員：

横矢 直和 教授	(主指導教員)
加藤 博一 教授	(副指導教員)
佐藤 智和 准教授	(副指導教員)
中島 悠太 助教	(副指導教員)

特徴点の明示的な対応付けを必要としない カメラ位置姿勢推定*

黒川 陽平

内容梗概

拡張現実感技術を用いたモバイル端末上で動作するアプリケーションでは、モバイル端末のカメラから得られる画像に対する情報の重畳提示のために、モバイル端末のカメラ位置姿勢を推定する必要がある。特に実時間性が要求されるこれらのアプリケーションにおいては、モバイル端末の限られたリソースを用いて高速・高精度なカメラ位置姿勢推定を実現することが要求される。従来、画像を用いた高精度なカメラ位置姿勢推定手法として、モバイル端末のカメラから得られる画像と事前に構築された3次元点群データベースを用いる手法が提案されている。この手法では、カメラで撮影された画像から検出した特徴点と点群データベースに含まれる点を記述子によって対応付けたうえで、Perspective-n-point (PnP) 問題を解くことで位置姿勢を推定する。しかし、この手法は画像上のそれぞれの特徴点について記述子を抽出し、さらに点群データベースに含まれる点に関連付けられた記述子との多数の組み合わせについてその類似度を算出する必要があることから計算コストが高い。また、多くの場合、記述子は高次元のベクトルであるため点群データベースのデータ量が増大し、ダウンロードや記憶コストの面でモバイル端末での利用には課題がある。

そこで本論文では、記述子による明示的な点の対応付けを伴わないカメラ位置姿勢推定手法を提案する。提案手法は、カメラから実時間で得られるクエリ画像から検出した特徴点の分布の偏りに着目し、データベース中の点群を画像平面に

*奈良先端科学技術大学院大学 情報科学研究科 情報科学専攻 修士論文, NAIST-IS-MT1351040, 2015年3月11日.

投影した投影点の分布とクエリ画像上の特徴点の分布の一致度合いに基づいて定義した尺度の最適化問題としてカメラ位置姿勢推定問題を帰着させる．具体的には，画像上の特徴点から真の位置姿勢における画像平面での投影点の分布を推定し，この分布に対してのある位置姿勢における投影点の当てはまりのよさを尺度として，その最大化によって位置姿勢を推定する．提案手法では，点の明示的な対応付けなしに位置姿勢を推定するため，点の対応付けのための記述子が不要である．そのため，従来手法に比べてデータベースサイズが小さく，省メモリなオンライン処理が実現できる．実験では，動画像を入力とした提案手法によるカメラ位置姿勢推定を推定精度・頑健性・計算負荷の観点で検証し，提案手法の有効性を示す．

キーワード

カメラ位置姿勢推定，自然特徴点，3次元点群データベース，拡張現実感

Camera Pose Estimation without Explicit Feature Matching*

Yohei Kurokawa

Abstract

Camera pose estimation is an essential technique for augmented reality applications on mobile devices in order to superimpose location specific information on an input image from their cameras. In such applications, camera poses must be estimated precisely in real-time within limited computational resources in mobile devices. Some precise methods use an image from the mobile device's camera (referred to as a query image) and preliminarily built 3D point cloud database. Such methods estimate camera poses by solving the perspective-n-point (PnP) problem with 2D-3D correspondences obtained by matching keypoints on a query image and points in the database using associated descriptors. However, this approach is computationally expensive because it extracts descriptors for every keypoint on the query image and makes correspondences from many possible combinations of keypoints and 3D points in the database. Additionally, the database size increases due to the descriptors each of which usually is a high dimensional vector. For these reasons, such a matching based method is not suitable for mobile device applications.

In this paper, we propose a novel camera pose estimation method without explicit keypoint matching. Observing the biased distribution of keypoints on the query image, and thus in the 3D point cloud database, we define a criterion that

*Master's Thesis, Department of Information Science, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-MT1351040, March 11, 2015.

gives the fitness of the distributions of the keypoints and the 3D point cloud projected to the image plane given a camera pose. Our proposed method formulates the problem of camera pose estimation as a maximization problem of this criterion. The advantage of our method is small storage for the database and memory usage compared with existing methods because it does not store descriptors. We experimentally demonstrate the accuracy, robustness, and computational cost of the proposed method.

Keywords:

Camera pose estimation, keypoints, 3D point cloud database, augmented reality

目次

1. はじめに	1
2. 従来研究と本研究の位置付け	3
2.1 センサベースのカメラ位置姿勢推定	3
2.2 画像ベースのカメラ位置姿勢	4
2.2.1 事前知識を用いない手法	4
2.2.2 事前知識を用いる手法	5
2.3 本研究の位置付けと方針	7
3. 特徴点の明示的な対応付けを伴わないカメラ位置姿勢推定	9
3.1 カメラ位置姿勢推定手法の概要	9
3.2 3次元点群データベースの構築	11
3.3 カメラ位置姿勢の評価尺度の定義	12
3.4 ルックアップテーブルによる評価尺度算出の高速化	16
3.5 評価尺度の最大化によるカメラ位置姿勢推定	18
4. 実験	20
4.1 実験条件	20
4.2 評価尺度の振る舞いの検証	22
4.3 カメラ位置姿勢推定の定量的評価	27
5. まとめ	32
謝辞	34
参考文献	35

目 次

1	Klein ら [22] による Visual SLAM	5
2	Kato ら [2] によるマーカを用いた拡張現実感のアプリケーション	6
3	Irschara ら [3] による 3次元点群データベースを用いたカメラ位置 姿勢推定	7
4	提案手法の流れ	10
5	提案手法で扱う 3次元点群データベース	11
6	特徴点と投影点	13
7	離散化されたスケールに対する寄与率	14
8	特徴点の分布 $d_k(\mathbf{p}, s)$ の例	15
9	ルックアップテーブルの生成	17
10	余白を設けない場合のルックアップテーブルの境界	17
11	3次元点群データベースの構築に用いた画像群の一部	20
12	実験に用いたカメラ画像	21
13	図 12(a) のカメラ画像に対する評価尺度の振る舞い	24
14	図 12(b) のカメラ画像に対する評価尺度の振る舞い	25
15	図 12(c) のカメラ画像に対する評価尺度の振る舞い	26
16	提案手法によるカメラ位置姿勢の推定結果	30
17	図 12(a) のカメラ画像に対するカメラ位置姿勢の初期値・推定値・ 真値における投影点の比較 (成功例)	31
18	図 12(a) のカメラ画像に対するカメラ位置姿勢の初期値・推定値・ 真値における投影点の比較 (失敗例)	31
19	図 12(a) のカメラ画像に対するルックアップテーブルの一部	31

表 目 次

1	カメラ位置姿勢推定手法の比較	8
---	--------------------------	---

1. はじめに

拡張現実感技術を用いたモバイル端末上で動作するアプリケーション(モバイル型拡張現実感)では, モバイル端末のカメラで撮影した実世界の動画像に対して視覚情報を重畳して提示するために, カメラの位置姿勢を高速かつ高精度に推定する必要がある. 拡張現実感のためのカメラ位置姿勢推定手法としてマーカを用いる手法 [1, 2] が広く利用されているが, 環境中にマーカを配置するため, 利用可能な環境が限定されるという問題がある.

マーカを用いずに高精度なカメラ位置姿勢推定を実現する手法として, 事前に構築した3次元点群データベースを用いる手法が提案されている [3, 4, 5, 6]. これらの手法は, カメラで撮影した画像(カメラ画像)から検出した特徴点と, 点群データベースに含まれる3次元点を対応付け, Perspective-n-Point 問題 [7, 8, 9] を解くことによってカメラ位置姿勢を推定する. 一般的に, 特徴点と3次元点の対応付けには点の周辺の見えに関する記述子が用いられる. この手法は, 高精度な推定が可能である一方で, カメラ画像からの記述子抽出や記述子による点の対応付けの計算コストが高く, リアルタイム性が要求されるモバイル型拡張現実感への適用は難しい. さらに, 点群データベースのすべての3次元点について対応付けのための記述子を保持する必要があるため, データベースのサイズが肥大し, モバイル端末での取り扱いが困難である.

このような問題に対して, 本論文ではリソースの限られたモバイル端末上でのマーカレス拡張現実感の実現に向けて, 記述子による特徴点の明示的な対応付けを伴わないカメラ位置姿勢推定手法を提案する. 具体的には, あるカメラ位置姿勢のもとで3次元点群を画像上に投影して得られる投影点と, 入力画像から検出した特徴点の, 画像上での位置およびスケールの一致度合いをカメラ位置姿勢の評価尺度とし, これを最大化することでカメラ位置姿勢を推定する. データベースには3次元点の位置とスケールのみを記録するためデータベースのサイズが小さく, モバイル端末での取り扱いが容易である.

以下, 2章では, カメラ位置姿勢推定の従来研究と本研究の位置付けについて述べる. 3章では, 本手法で用いる3次元点群データベースの構築, カメラ位置姿勢の評価尺度, 最大化によるカメラ位置姿勢の推定手法について詳述する. 4

章では，本研究で提案するカメラ位置姿勢の評価尺度の振る舞いを検証した上で，提案手法によるカメラ位置姿勢推定を定量的に評価する．最後に5章でまとめと今後の課題について述べる．

2. 従来研究と本研究の位置付け

本章では、拡張現実感で用いられるカメラ位置姿勢推定の従来研究を、センサベースの手法、事前知識を用いない画像ベースの手法、事前知識を用いた画像ベースの手法、それぞれの手法の特徴とモバイル型拡張現実感への適用可能性について述べる。また、本研究の位置付けと方針について述べる。

2.1 センサベースのカメラ位置姿勢推定

センサベースのカメラ位置姿勢推定手法には、環境内インフラとしてユーザの位置姿勢が計測できるセンサを設置する手法 [10, 11] と、モバイル端末に搭載された GPS や電子コンパスなどのセンサを利用する手法 [12, 13, 14, 15, 16] がある。

インフラを利用する手法として、Newman ら [10] は、ユーザが信号の発信機を複数装備し、環境内に設置された多数の受信機で信号を捉え、それにかかる時間を計測する (Times-of-Flight) ことで、ユーザの位置姿勢を計測する手法を提案した。また、Tenmoku ら [11] は、環境内に赤外線ビーコンと RFID を埋め込み、歩数計を補助的に用いることで、ユーザの位置姿勢を計測する手法を提案した。このような手法は、モバイル端末の計算リソースを圧迫しないという利点があるが、環境インフラの整備やユーザの装備が大がかりになるという問題があり、様々な環境においてモバイル型拡張現実感が実現できるとは言い難い。

モバイル端末に搭載されたセンサを用いる手法 [12, 13, 14, 15, 16] では、一般的に、GPS を用いて端末の位置を、電子コンパスと加速度センサを用いて端末の姿勢を計測する。この手法は、環境インフラの整備が不要で、計算コストも低く、絶対的な位置姿勢が取得できるという利点があり、インフラを用いる手法に比べてモバイル型拡張現実感への適用可能性が高い。しかし、モバイル端末とセンサのキャリブレーション誤差や計測誤差に起因して、得られる位置姿勢の精度が低いという問題があり、拡張現実感に求められる画素単位の位置合わせが難しい。

2.2 画像ベースのカメラ位置姿勢

画像ベースのカメラ位置姿勢推定手法は，入力画像から得られる特徴点やエッジなどを追跡，もしくはそれらを環境に関する事前知識と対応付けることでカメラ位置姿勢を推定する．この手法では一般的に，追跡や事前知識との対応付けを画素単位，またはそれより小さな単位で行うため，センサベースの手法に比べてカメラ位置姿勢の推定精度を高めやすい．以下では，画像ベースのカメラ位置姿勢推定手法を事前知識を用いない手法 [17, 18, 19, 20, 21, 22, 23, 24, 25] と事前知識を用いた手法 [1, 2, 3, 4, 5, 6, 26, 27] に分け，それぞれの手法について概観する．

2.2.1 事前知識を用いない手法

事前知識を用いず，画像群のみからカメラ位置姿勢を推定する手法には，Simultaneous Localization and Mapping (SLAM) [17, 18, 19, 20, 21, 22, 23, 24, 25] がある．SLAM は，未知の環境において，自己位置姿勢の推定と環境マップの構築を同時に行う手法である．SLAM には，レーザースキャナなどを用いて得られる環境情報に基づく手法 [17, 18, 19] と，カメラを用いて環境を撮影することで得られる環境の視覚情報に基づく手法 (Visual SLAM) [20, 21, 22, 23, 24, 25] に大別できる．ここでは，モバイル型拡張現実感への適用を考え，多くのモバイル端末が備えるカメラを利用する Visual SLAM について述べる．Visual SLAM は，カメラ画像から特徴点を検出・追跡することで，カメラ位置姿勢推定と環境マップの構築を同時に行う．この手法を拡張現実感に適用するには，カメラ位置姿勢の推定，特徴点の検出・追跡，環境マップの構築・更新をリアルタイムに実行する必要があり，計算コストの面で課題があった．これに対し Klein ら [22, 23] は，カメラ位置姿勢推定と環境マップ構築を並列化し，環境マップの構築・更新にかかる計算時間がカメラ位置姿勢推定のリアルタイム性に影響を及ぼすことを防いだ手法を提案した (図 1)．しかし，SLAM は推定誤差が蓄積するため，小さな環境への適用に限定されるという問題がある．また，SLAM によって推定されるカメラ位置姿勢は，動的に構築された環境マップに対する相対的なものであるため，

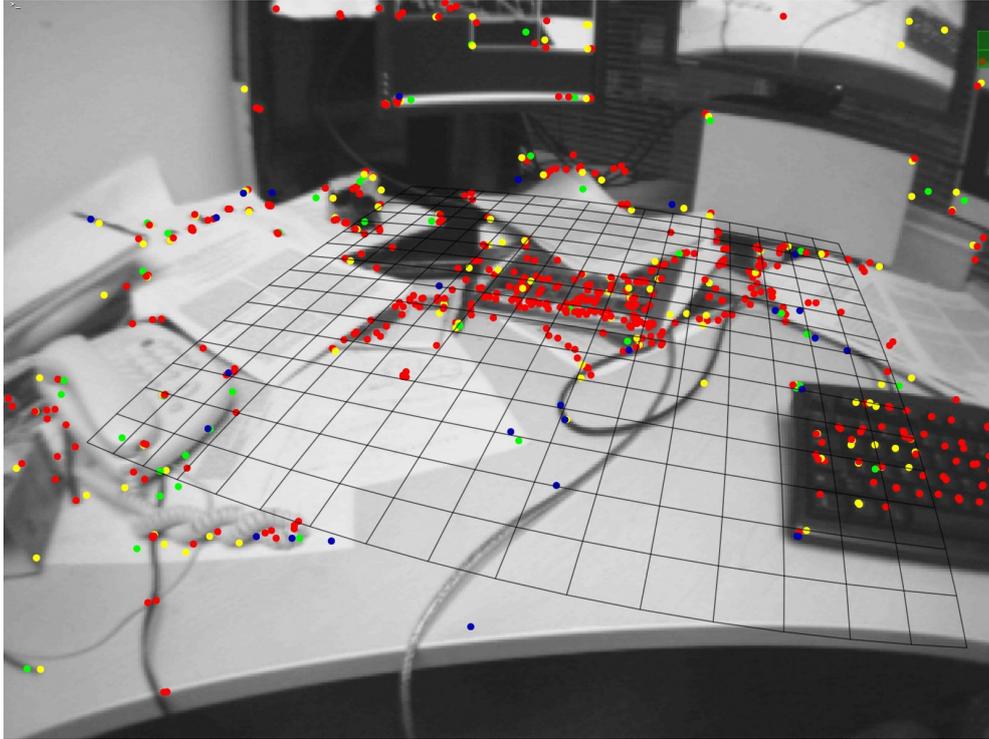


図 1: Klein ら [22] による Visual SLAM

環境中の位置に応じて視覚情報を重畳提示するような拡張現実感は単体では実現できない。こうした問題を回避するには、絶対的な指標となるセンサや事前知識を併用する必要がある。そこで、Castle ら [24] は Klein ら [22] の手法を拡張し、事前に構築した環境マップと動的に構築される環境マップを対応付けることで環境中の位置に応じた視覚情報の重畳提示を可能にした。しかしこの手法は、主に机上等の小さな空間を対象としており、日時や天候による見えの変化を伴う屋外環境での利用については言及していない。

2.2.2 事前知識を用いる手法

事前知識を用いる手法は、画像から得られる特徴点などの情報を事前知識と対応付けることでカメラ位置姿勢を推定する。これらの手法では、事前知識として、マーカ [1, 2], 3次元 CAD モデル [26], および 3次元点群 [3, 4, 5, 6, 27] などが用いられる。

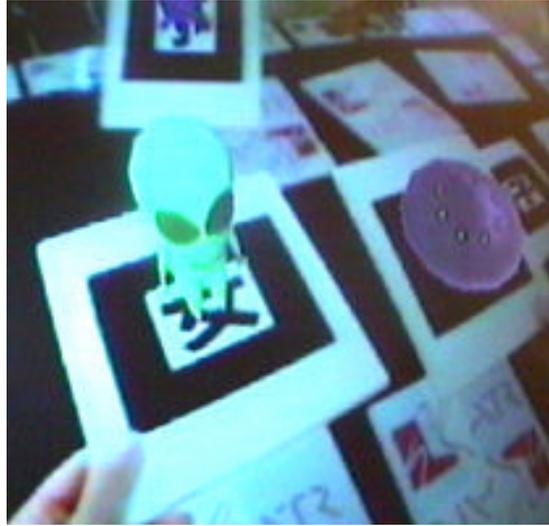


図 2: Kato ら [2] によるマークを用いた拡張現実感のアプリケーション

マークを用いる手法は、画像上での検出が容易なマークを環境中に配置し、マークの 3 次元座標と画像中のマークの 2 次元座標を対応付けることで、マークを基準としたカメラ位置姿勢を推定する [1, 2] (図 2)。この手法はリアルタイムにカメラ位置姿勢を推定できるため、拡張現実感のアプリケーションで広く用いられている。しかし、環境中にマークを配置する必要があるため、利用可能な環境が限定されるという問題がある。

3 次元 CAD モデルを用いる手法は、環境に関する 3 次元 CAD モデルをあるカメラ位置姿勢のもとで投影し、CAD モデルの線分とカメラ画像から検出したエッジの距離を最小化することでカメラ位置姿勢を推定する [26]。この手法は、マークを用いることなく高精度にカメラ位置姿勢を推定できるが、対象環境が 3 次元 CAD モデルとして表現可能な環境に限られ、さらに 3 次元 CAD モデルの構築に専門的な知識や技能を要する。

3 次元点群を用いる手法は、Structure from Motion 法 [28, 29, 30, 31] などを利用して事前に構築した 3 次元点群データベースに含まれる 3 次元点とカメラ画像から検出した特徴点を対応付けることでカメラ位置姿勢を推定する [3, 4, 5, 6, 27] (図 3)。点の対応付けには、一般的に、特徴点の見えに関する記述子が用いられる。この手法も、マークを配置せずに高精度なカメラ位置姿勢が推定できる。し

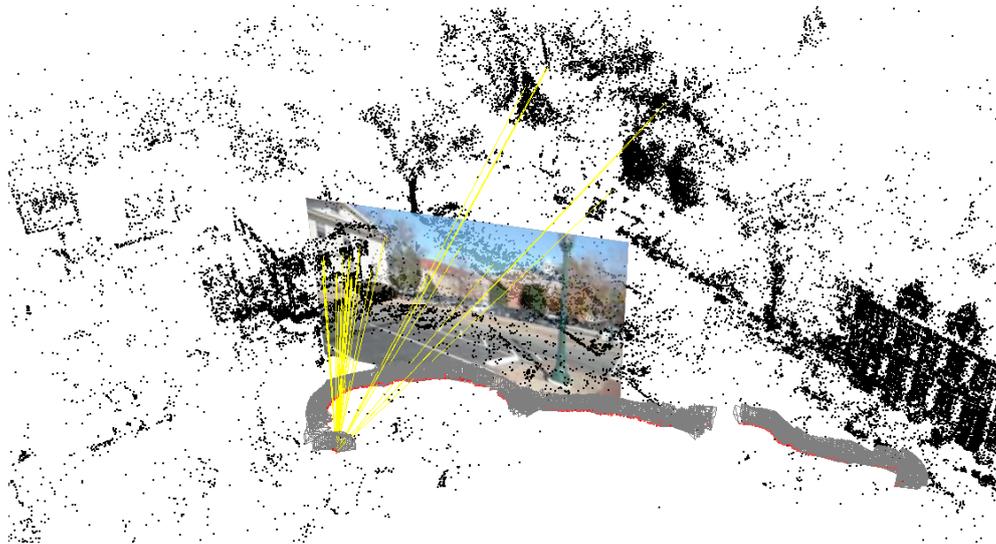


図 3: Irschara ら [3] による 3 次元点群データベースを用いたカメラ位置姿勢推定

かし、画像からの記述子抽出や高次元の記述子による点の対応付けの計算コストが高く、リアルタイム性が要求される拡張現実感への適用には課題があった。これに対し、Taketomi ら [27] は、環境のランドマークデータベースを用いて、優先度に基づくランドマークの対応付けとランドマークの追跡により計算コストを削減し、高速・高精度にカメラ位置姿勢を推定する手法を提案した。しかし、こうしたデータベースを用いる手法では、各 3 次元点ごとに対応付けのための記述子やランドマークを格納しておく必要があるためデータベースが肥大化し、依然としてモバイル端末での取り扱いが難しい。

2.3 本研究の位置付けと方針

前節までに概観したように、カメラ位置姿勢を推定する手法は数多く提案されている。それらの手法の特徴を表 1 にまとめる。モバイル型拡張現実感では、高速かつ高精度にカメラ位置姿勢を推定することが求められるため、この点では Visual SLAM が適する。しかしながら、Visual SLAM は動的に構築される環境マップに対してカメラ位置姿勢を推定するため、対象環境の所定の箇所に視覚情報を重畳するには、センサや事前知識を併用する必要がある。センサを用いる手

表 1: カメラ位置姿勢推定手法の比較

手法	計算コスト	精度	拡張現実感への適用にあたっての問題点
センサベース	低	低	精度が低い
Visual SLAM	低	高	絶対的な位置姿勢が得られない, 誤差が累積する
3次元点群	高	高	計算コストが高い, データベースサイズが大きい

法は、リソースの限られたモバイル端末でも高速に動作するが、推定精度の面で問題があることから、精度を高めるためには3次元点群などの事前知識が必要である。

そこで本研究では、拡張現実感に要求される高精度なカメラ位置姿勢推定を実現する手法として、3次元点群データベースを用いる手法に着目する。この手法は計算コストとデータベースサイズに関して課題があるが、これは記述子を利用することに起因する。この課題を解決するため、提案手法では、記述子による明示的な特徴点の対応付けを伴わずにカメラ位置姿勢を推定する。具体的には、あるカメラ位置姿勢のもとで3次元点群を画像上に投影して得られる投影点と、入力画像から検出した特徴点の、画像上での位置およびスケールの一致度合いをカメラ位置姿勢の評価尺度とし、これを最大化することでカメラ位置姿勢を推定する。これによって、カメラ画像上の特徴点に対する記述子の抽出が不要となり、さらにデータベースに3次元点に関する記述子を保存する必要がないため、モバイル端末上での取り扱いが容易になる。

3. 特徴点の明示的な対応付けを伴わないカメラ位置姿勢推定

本章では、まず提案手法の概要を述べ、続いて3次元点群データベースの構築法について概説した上で、カメラ位置姿勢推定に用いる各要素について詳述する。

3.1 カメラ位置姿勢推定手法の概要

図4に提案手法全体の流れを示す。提案手法では、まずオフライン処理として、対象環境の画像群から Structure from Motion 法 [28, 29, 30, 31] によって3次元点群データベースを構築する。3次元点群データベース中の各3次元点には、3.2節で述べる方法によって3次元点スケールを付与する。オンライン処理では、リアルタイムに取得されるカメラ画像からスケールを持つ特徴点(例えば SIFT [32] や SURF [33])を検出し、画像平面上の特徴点の分布に関するルックアップテーブル(LUT)を生成する。続いて、カメラの位置 (t_x, t_y, t_z) と姿勢 (r_x, r_y, r_z) の計6パラメータ $\theta = (r_x, r_y, r_z, t_x, t_y, t_z)$ と既知の内部パラメータ $\{f, c_x, c_y\}$ により3次元点群を画像平面に投影し、すべての投影点についてルックアップテーブルを参照することでカメラ位置姿勢に関する評価尺度を計算する。この評価尺度を勾配法に基づいて最大化することによりカメラ位置姿勢を推定する。提案手法では、特徴点のスケールを考慮した分布を導入することにより、真に対応しない特徴点と投影点の位置の一致により評価尺度の値が大きくなることを防ぐ。

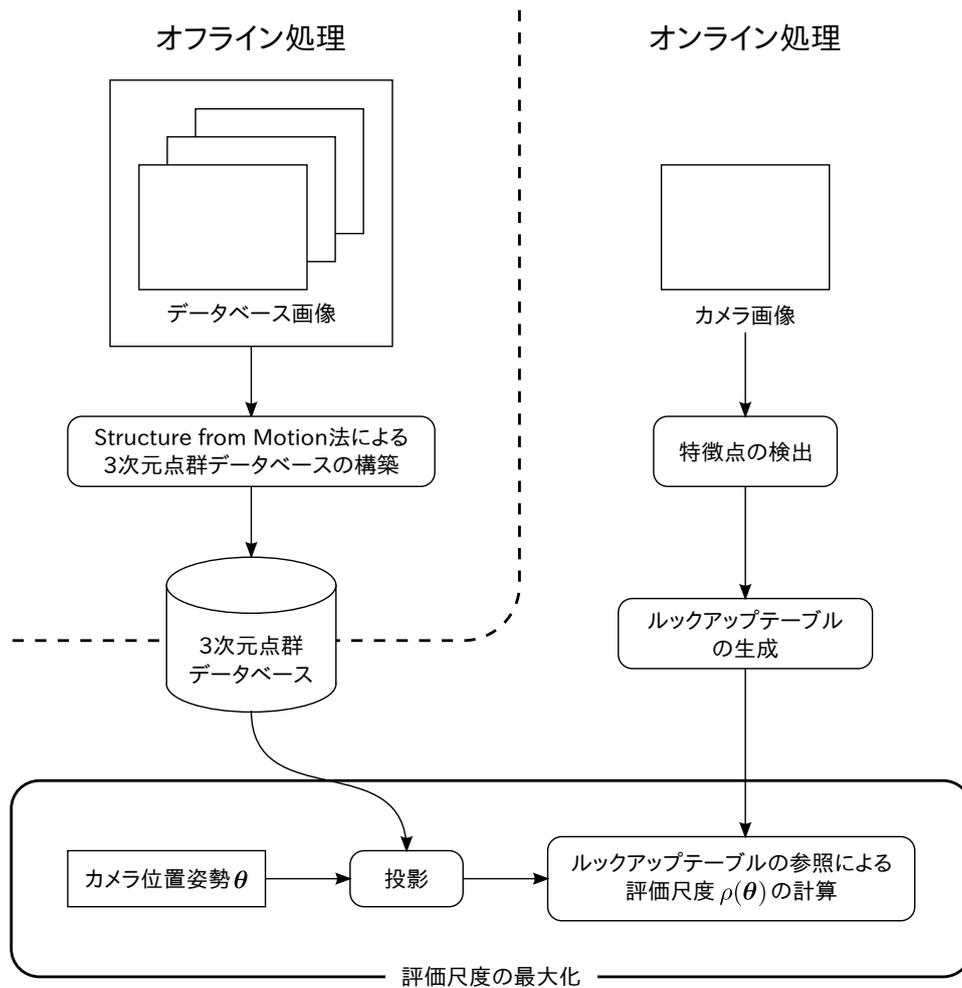


図 4: 提案手法の流れ

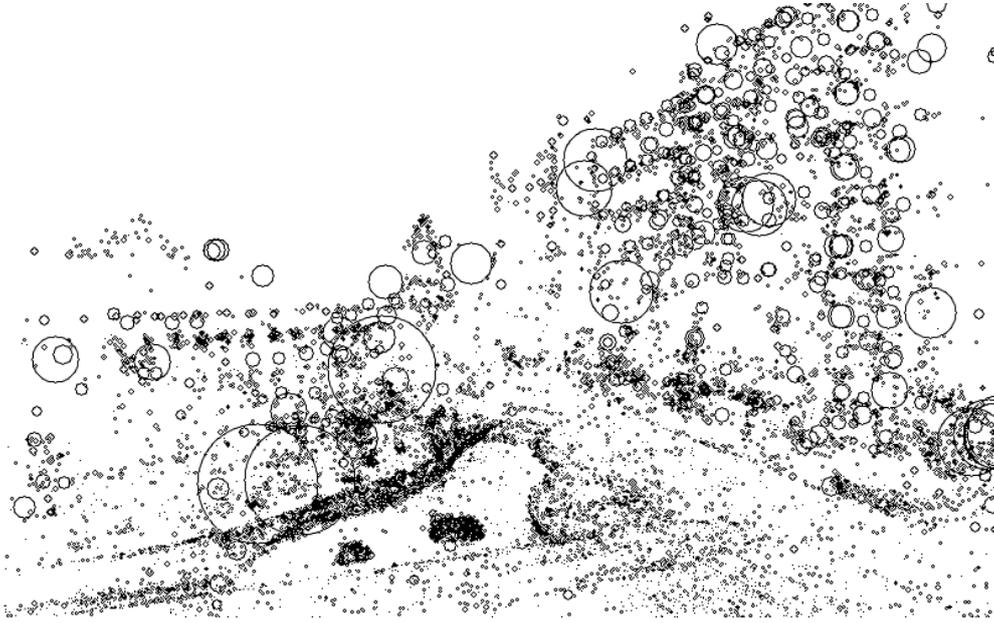


図 5: 提案手法で扱う 3次元点群データベース

3.2 3次元点群データベースの構築

提案手法で扱う 3次元点群データベースは、点群 $\mathcal{P} = \{\mathbf{P}_j | j = 1, \dots, J\}$ と、それぞれの点 $\mathbf{P}_j = (X_j, Y_j, Z_j)$ に対応する 3次元点スケール S_j から構成される。図 5 に 3次元点群データベースの例を示す。図中の円の中心が 3次元点位置 (X_j, Y_j, Z_j) を、円の大きさが 3次元点スケール S_j を表している。点群 \mathcal{P} は、事前に撮影された対象環境の画像群 (データベース画像) を入力として、Structure from Motion (SFM) 法により得られる。SFM 法は、多くの場合、入力画像群中の画像 D_n ($n = 1, \dots, N$) から特徴点とそれぞれの特徴点に対する記述子を算出し、記述子による特徴点の対応付けに基づいて、3次元点群 \mathcal{P} と入力画像群を撮影したカメラの位置姿勢を求める。このとき、点 \mathbf{P}_j が検出された入力画像 D_n 中の特徴点の固有スケール $s_{n,j}$ は、理想的にはどの方向・距離で撮影された画像上であっても、点 \mathbf{P}_j を中心とする同一範囲の物体を含む画像上の領域を表している。この画像上の領域に対応する 3次元空間中の領域を、点 \mathbf{P}_j を中心とする半径 S の球で表せば、一般に、焦点距離 f のカメラから奥行き z の位置にある半径 S の球が、画像上で半径 s の円として観測された場合、その関係は $S = zs/f$ で

与えられる。点 \mathbf{P}_j は複数の入力画像で観測されることから，提案手法では点 \mathbf{P}_j に対する3次元点スケール S_j を，複数画像のスケール $s_{n,j}$ に基づく値の平均として，次式により定義する。

$$S_j = \frac{1}{N_j} \sum_n \frac{z_{n,j} s_{n,j}}{f_n} \quad (1)$$

ここで， $z_{n,j}$ は画像 D_n を撮影したカメラ中心から \mathbf{P}_j までの奥行き， f_n は画像 D_n を撮影したカメラの焦点距離， N_j は \mathbf{P}_j を観測した入力画像数であり，和はこれらの画像に対して計算される。得られた3次元点群 \mathcal{P} と対応する3次元点スケール $\mathcal{S} = \{S_j | j = 1, \dots, J\}$ をデータベースに保存する。

3.3 カメラ位置姿勢の評価尺度の定義

提案手法では，図6に示すように，カメラ画像(図6(a))から検出される特徴点(図6(b))と，あるカメラ位置姿勢のもとで3次元点群を投影して得られる投影点(図6(c))の，画像平面上における分布の偏りに着目する。この偏りによって，カメラ位置姿勢に応じて，投影点と特徴点の分布の一致度合いに差が生じる。そこで，投影点と特徴点の分布の一致度合いに関する評価尺度をカメラ位置姿勢の関数として定義し，その尺度を最大化するカメラ位置姿勢を探索することによりカメラ位置姿勢を推定する。

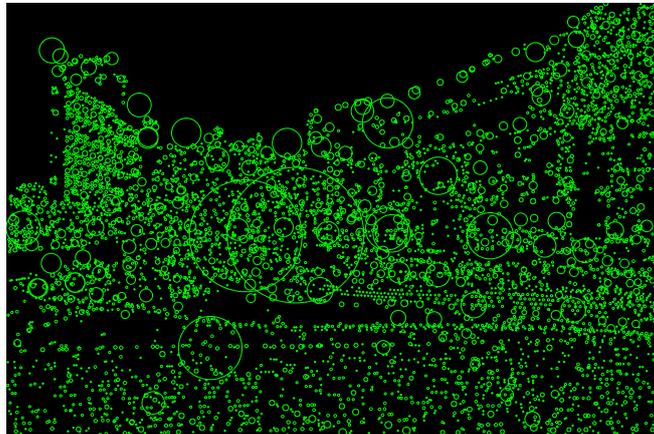
提案手法はまず，リアルタイムに得られるカメラ画像からスケールを持つ特徴点を検出する。得られた特徴点集合を $\mathcal{Q} = \{\mathbf{q}_i | i = 1, \dots, I\}$ とし，それぞれの特徴点に座標 $\mathbf{q}_i = (u_i, v_i)$ とスケール σ_i が与えられる。提案手法では，連続値として得られるスケールの値を K 段階に離散化し，それぞれのスケール Σ_k ($k = 1, \dots, K$) における特徴点の分布を考える。このとき，それぞれについて特徴点 \mathbf{q}_i が σ_i に応じて寄与するものと考え，画像上のある位置 \mathbf{p} におけるスケール Σ_k での特徴点の分布 $d_k(\mathbf{p})$ を，ガウシアンカーネル $g(\mathbf{x}) = \exp(-\beta \mathbf{x}^\top \mathbf{x})$ を用いて次式により表す。

$$d_k(\mathbf{p}) = \sum_{i=1}^I w_k(\sigma_i) g(\mathbf{p} - \mathbf{q}_i) \quad (2)$$

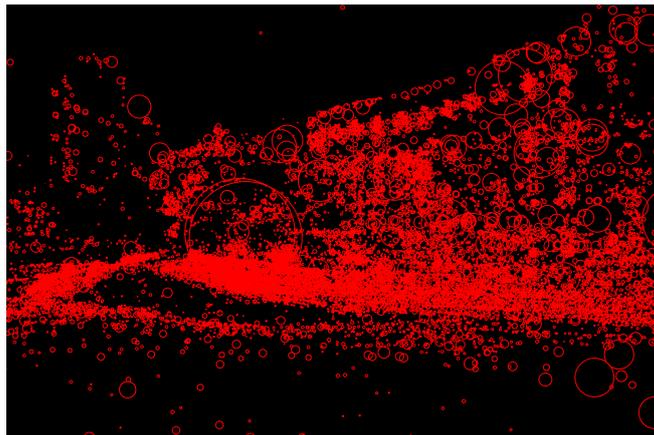
また，点 \mathbf{p} のスケールを s としたとき， \mathbf{p} における特徴点の分布は σ_i と同様に s に応じた寄与率によって与えられるものと考え，次式により d_k の各スケールを



(a) カメラ画像



(b) 特徴点



(c) 投影点

図 6: 特徴点と投影点

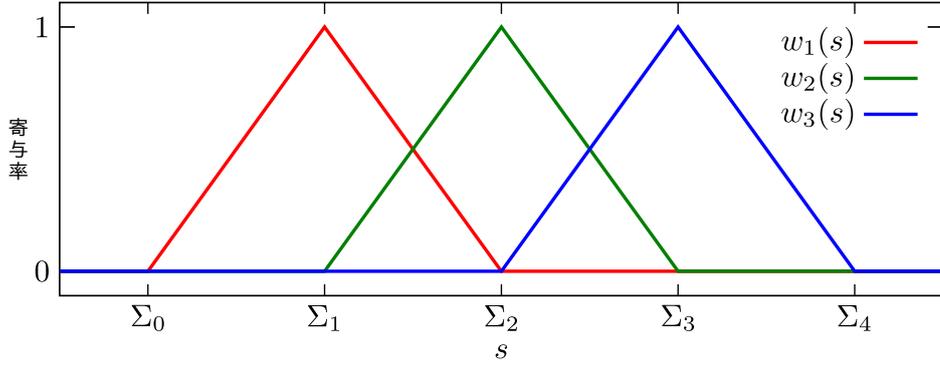


図 7: 離散化されたスケールに対する寄与率

統合する.

$$d(\mathbf{p}, s) = \sum_{k=1}^K w_k(s) d_k(\mathbf{p}, s) \quad (3)$$

ただし, $w_k(\bullet)$ は離散化されたスケール Σ_k に対する s , σ_i の寄与率を表す重みである. 提案手法では, スケールを図7に示すように離散化し, 線形的に変化する重みを次式により設定する.

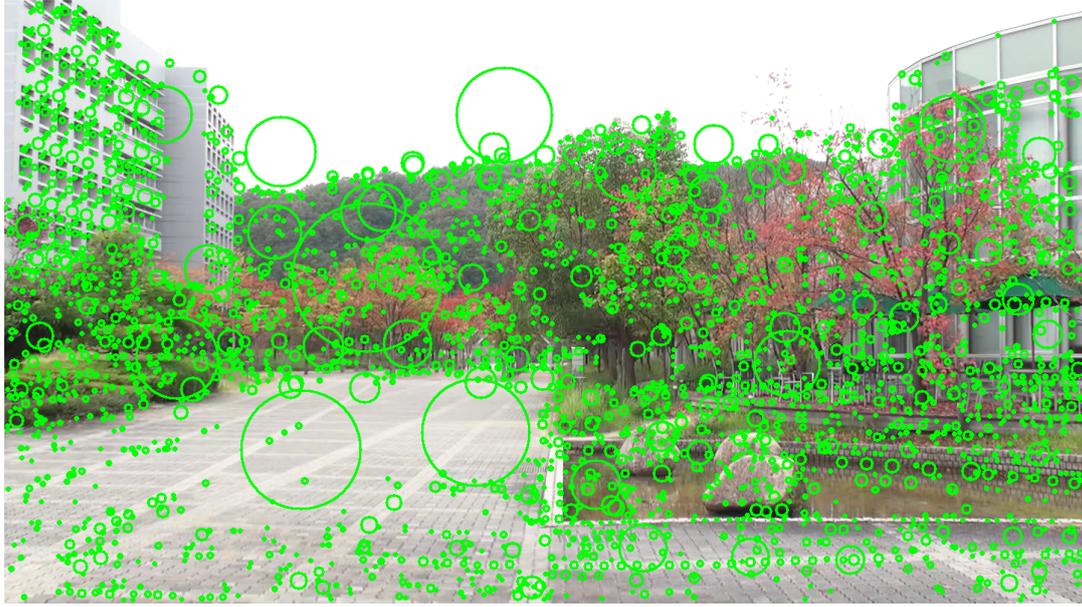
$$w_k(s) = \begin{cases} \max\left(1 - \frac{\Sigma_k - s}{\Sigma_k - \Sigma_{k-1}}, 0\right) & (s \leq \Sigma_k) \\ \max\left(1 - \frac{s - \Sigma_k}{\Sigma_{k+1} - \Sigma_k}, 0\right) & (s > \Sigma_k) \end{cases} \quad (4)$$

図8に示すように, 特徴点の分布 d は対象環境の3次元形状やテクスチャに依存し, 特徴点の密度が高いところほど大きな値となる. また, β を変更することで, 特徴点の分布をどの程度の粒度で表すかを調整できる.

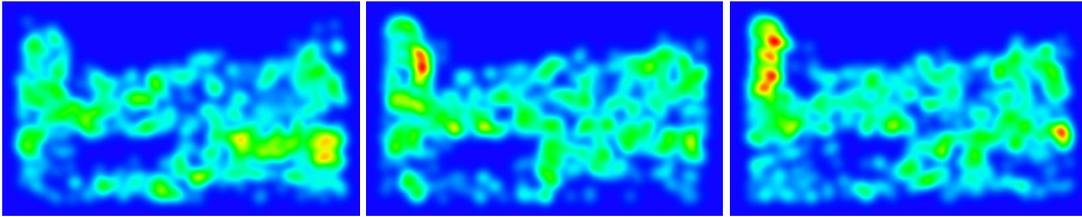
任意のカメラ位置姿勢 θ が与えられたとき, そのカメラ位置姿勢のもとで得られる投影点とカメラ画像から検出される特徴点の分布の一致度合いを評価する尺度 ρ は, $d(\mathbf{p}, s)$ を利用して次式により定義する.

$$\rho(\theta) = \sum_{j=1}^J d(\text{proj}_{\theta}^{\mathbf{p}}(\mathbf{P}_j), \text{proj}_{\theta}^{\mathbf{s}}(S_j)) \quad (5)$$

ただし, $\text{proj}_{\theta}^{\mathbf{p}}(\mathbf{P}_j)$, $\text{proj}_{\theta}^{\mathbf{s}}(S_j)$ は, 既知のカメラ内部パラメータ $\{f, c_x, c_y\}$ とカメラ位置姿勢 θ による3次元点 $\mathbf{P}_j = (X_j, Y_j, Z_j)$ と3次元点スケール S_j の画像平



(a) 検出された特徴点



(b) d_1

(c) d_2

(d) d_3

図 8: 特徴点の分布 $d_k(\mathbf{p}, s)$ の例

面への投影であり，次式で与えられる．

$$\text{proj}_{\theta}^{\mathbf{p}}(\mathbf{P}_j) = \begin{pmatrix} fx_j/z_j + c_x \\ fy_j/z_j + c_y \end{pmatrix} \quad (6)$$

$$\text{proj}_{\theta}^s(S_j) = fS_j/z_j \quad (7)$$

$$\begin{pmatrix} x_j \\ y_j \\ z_j \end{pmatrix} = \mathbf{R}(\theta) \begin{pmatrix} X_j \\ Y_j \\ Z_j \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \quad (8)$$

ただし， $\mathbf{R}(\theta)$ は θ によって決定される回転行列である．式 (5) により定義される評価尺度 $\rho(\theta)$ は， θ による投影点と特徴点の分布の一致度合いが高いとき大きな

値となる。

3.4 ルックアップテーブルによる評価尺度算出の高速化

提案手法では、式 (5) を最大とする θ を求めることで、カメラの位置姿勢を推定する。しかし、式 (5) は異なる θ での評価尺度の算出に KIJ 回のガウシアンカーネルの計算が必要となり、 I は数千、 J は数万から数十万点に及ぶことから、モバイル端末上でリアルタイムに動作させるには計算コストが高い。そこで、離散化したスケールにおける特徴点の分布 d_k をルックアップテーブルとして保持する。ルックアップテーブルはカメラ画像を取得するごとに 1 回のみ生成され、これにより KJ 回のテーブル参照によって評価尺度の近似値が得られる。

図 9 にルックアップテーブルの生成の流れを示す。提案手法では、ルックアップテーブルとして、カメラ画像と同じ大きさのテーブル (図 9(a)) の上下左右に十分な余白を加えたものを用意する (図 9(b))。次に、図 9(c) に示すように、すべての \mathbf{q}_i についてルックアップテーブル上の対応するセルの値を設定する。このとき、 \mathbf{q}_i は一般に整数値ではないことから、バイリニア補間と同様の考え方により、 \mathbf{q}_i の周囲の 4 つのセル値を \mathbf{q}_i の値に応じて合計が $w_k(s_i)$ になるように設定する。最後に、 $g(\mathbf{x})$ をカーネルとするガウシアンフィルタを適用する (図 9(d))。以上の手順により得られるルックアップテーブルを $\text{LUT}_k(\mathbf{p})$ とする。これを用いて $d(\mathbf{p}, s)$ を

$$d'(\mathbf{p}, s) = \sum_{k=1}^K w_k(s) \text{LUT}_k(\mathbf{p}) \quad (9)$$

により近似する。 \mathbf{p} が非整数値の場合、 $\text{LUT}_k(\mathbf{p})$ はバイリニア補間により対応する値を与えるものとする。ここで、特徴点 \mathbf{q}_i は画像上の任意の位置で検出されることから、余白を加えない場合、図 10 に示すように画像の境界において値を持つ可能性がある。このとき、最適化の過程で境界付近の投影点が θ の変化によって境界を逸脱すると、評価尺度が不連続に変化する可能性がある。提案手法では、カーネルサイズに応じた十分な大きさの余白を導入することによってこの不連続を解消する。

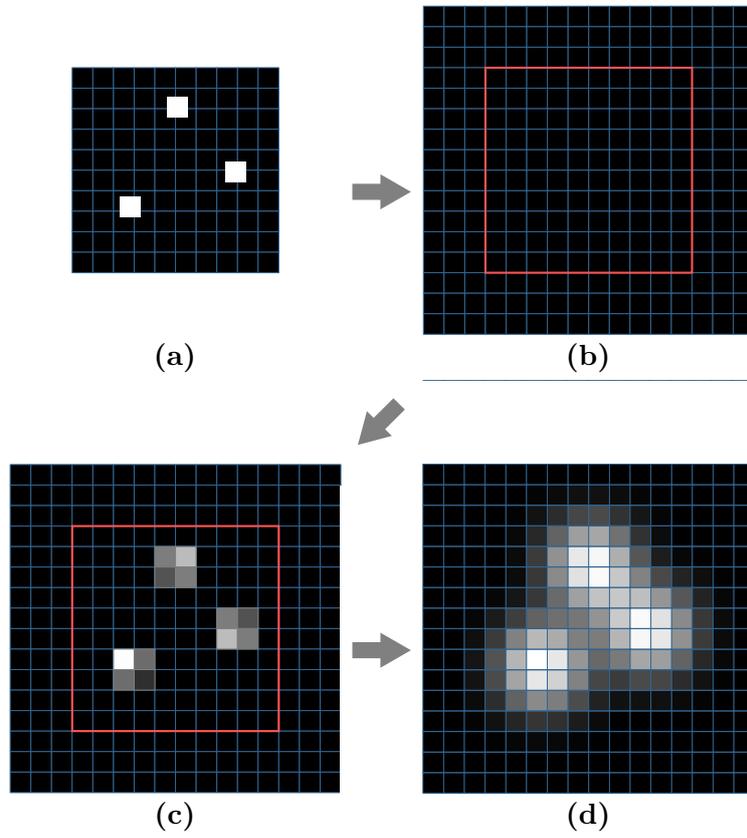


図 9: ルックアップテーブルの生成

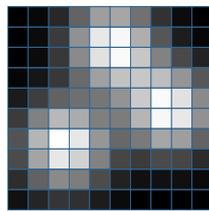


図 10: 余白を設けない場合のルックアップテーブルの境界

3.5 評価尺度の最大化によるカメラ位置姿勢推定

前節で述べた手法で生成したルックアップテーブルを用いて、勾配法に基づき評価尺度 $\rho(\boldsymbol{\theta})$ を $\boldsymbol{\theta}$ に関して最大化することでカメラ位置姿勢を推定する。

提案手法では、式 (5) に示す通り、 $\rho(\boldsymbol{\theta})$ をルックアップテーブルにより近似的に計算するため、 $\rho(\boldsymbol{\theta})$ の勾配を解析的に算出することができない。そこで、差分近似により勾配を算出する。 $\boldsymbol{\theta} = (r_x, r_y, r_z, t_x, t_y, t_z)$ の各成分に関する $\rho(\boldsymbol{\theta})$ の微分の差分近似は次のように定義される。

$$\begin{aligned}
 \frac{\partial}{\partial r_x} \rho(\boldsymbol{\theta}) &\approx \frac{\rho(\boldsymbol{\theta} + \boldsymbol{\theta}_{\Delta r_x}) - \rho(\boldsymbol{\theta})}{\Delta r_x} \\
 \frac{\partial}{\partial r_y} \rho(\boldsymbol{\theta}) &\approx \frac{\rho(\boldsymbol{\theta} + \boldsymbol{\theta}_{\Delta r_y}) - \rho(\boldsymbol{\theta})}{\Delta r_y} \\
 \frac{\partial}{\partial r_z} \rho(\boldsymbol{\theta}) &\approx \frac{\rho(\boldsymbol{\theta} + \boldsymbol{\theta}_{\Delta r_z}) - \rho(\boldsymbol{\theta})}{\Delta r_z} \\
 \frac{\partial}{\partial t_x} \rho(\boldsymbol{\theta}) &\approx \frac{\rho(\boldsymbol{\theta} + \boldsymbol{\theta}_{\Delta t_x}) - \rho(\boldsymbol{\theta})}{\Delta t_x} \\
 \frac{\partial}{\partial t_y} \rho(\boldsymbol{\theta}) &\approx \frac{\rho(\boldsymbol{\theta} + \boldsymbol{\theta}_{\Delta t_y}) - \rho(\boldsymbol{\theta})}{\Delta t_y} \\
 \frac{\partial}{\partial t_z} \rho(\boldsymbol{\theta}) &\approx \frac{\rho(\boldsymbol{\theta} + \boldsymbol{\theta}_{\Delta t_z}) - \rho(\boldsymbol{\theta})}{\Delta t_z}
 \end{aligned} \tag{10}$$

ここで、 $\boldsymbol{\theta}_{\Delta r_x}$ から $\boldsymbol{\theta}_{\Delta t_z}$ は $\boldsymbol{\theta}$ を各軸方向に微小に変化させるベクトルであり、次のように定義される。

$$\begin{aligned}
 \boldsymbol{\theta}_{\Delta r_x} &= (\Delta r_x, 0, 0, 0, 0, 0) \\
 \boldsymbol{\theta}_{\Delta r_y} &= (0, \Delta r_y, 0, 0, 0, 0) \\
 \boldsymbol{\theta}_{\Delta r_z} &= (0, 0, \Delta r_z, 0, 0, 0) \\
 \boldsymbol{\theta}_{\Delta t_x} &= (0, 0, 0, \Delta t_x, 0, 0) \\
 \boldsymbol{\theta}_{\Delta t_y} &= (0, 0, 0, 0, \Delta t_y, 0) \\
 \boldsymbol{\theta}_{\Delta t_z} &= (0, 0, 0, 0, 0, \Delta t_z)
 \end{aligned} \tag{11}$$

式 (10) に基づいて算出した 6 次元の勾配ベクトル $\nabla \boldsymbol{\theta} = \left(\frac{\partial}{\partial r_x} \rho(\boldsymbol{\theta}), \dots, \frac{\partial}{\partial t_z} \rho(\boldsymbol{\theta}) \right)$ をもとに評価尺度を最大化する。最大化におけるパラメータの更新は次の式に

従う.

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha \nabla \boldsymbol{\theta} \quad (12)$$

ただし, α は更新量を調整する小さな正の実数であり, $\rho(\boldsymbol{\theta} + \alpha \nabla \boldsymbol{\theta})$ を最大化するよう直線探索によって動的に決定する. この更新を $\rho(\boldsymbol{\theta})$ が収束するまで繰り返す, 最適なカメラ位置姿勢を推定する.

4. 実験

本研究では、評価尺度の有用性を示すために評価尺度の振る舞いを検証する実験と、提案手法によるカメラ位置姿勢推定の精度を定量的に明らかにする実験を行った。本章では、それらの実験の内容と結果の詳細を述べ、考察する。

4.1 実験条件

3次元点群データベースの構築のために、本研究では同一のカメラから撮影された160枚の画像(1,920×1,280画素)から構成される画像群(データベース画像)を用いた。図11にデータベース画像の一部を例示する。3次元点群データベースの構築、およびカメラの内部パラメータ推定にはVisualSFM [34]を用いた。得られた3次元点群(図5)は22,147点で、復元された点群の座標系においては実世界の1mが並進成分のおよそ0.4に相当する。3次元点位置と3次元点スケールが



図 11: 3次元点群データベースの構築に用いた画像群の一部



(a)



(b)



(c)

図 12: 実験に用いたカメラ画像

ら構成されるデータベース全体のサイズは 346 kB であった。また、位置姿勢推定の対象となるカメラ画像として、データベース画像を撮影したカメラとは異なるカメラを用いて異なる時期に撮影した 3 枚の画像 ($1,440 \times 810$ 画素) を用いた。図 12 にこれらの画像を示す。カメラ画像からの特徴点検出には SiftGPU [35] を利用した。これらのカメラ画像の位置姿勢 θ の真値は、点群データベース構築時に用いた特徴点の 3 次元位置を画像上の点に対応付けることで推定した。

式 (2) で用いられるガウシアンカーネルのパラメータ β は、カメラ画像の大きさを目安に経験的に 0.05 に設定した。ルックアップテーブルの余白は、ガウシアンフィルタの結果が画像内に収まるよう、 β に応じて設定した。

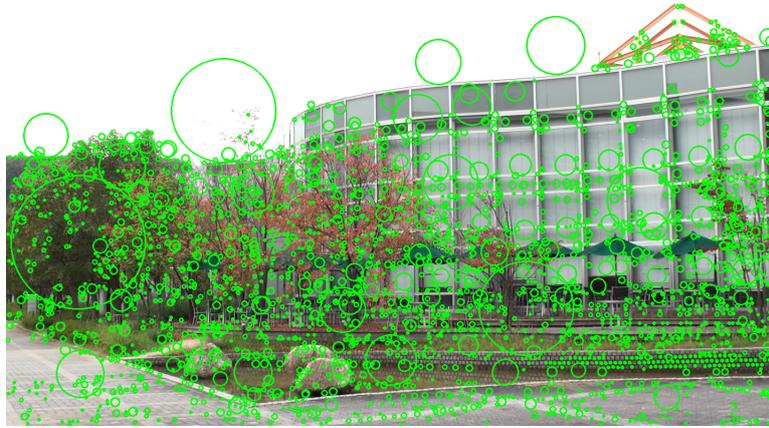
ルックアップテーブルを構成する画像の幅・高さは、カメラ画像の幅・高さにルックアップテーブルの余白を加えた大きさとなる。カメラ画像が高解像度である場合、ルックアップテーブルも高解像度となり、メモリ使用量の増加、ルックアップテーブル生成コストの増加、CPU キャッシュヒット率の低下といった問題を引き起こす。そこで、実装上の効率化として、必要に応じてルックアップテーブルを縮小する。ルックアップテーブル画像を縦横それぞれ A 倍する場合、特徴点の座標とスケール、内部パラメータもそれぞれ A 倍する。

4.2 評価尺度の振る舞いの検証

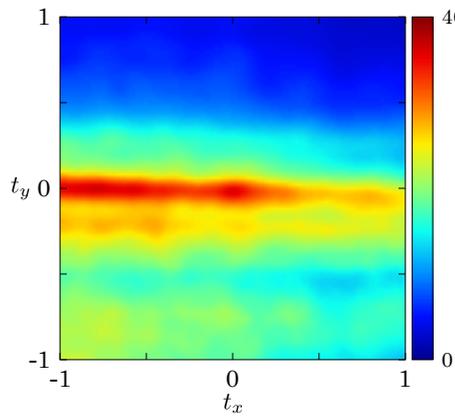
式 (5) で提案した評価尺度 $\rho(\theta)$ が真値において最大値となることを確認するために、パラメータ θ の変化に応じて $\rho(\theta)$ がどのように振る舞うかを検証する実験を行った。本実験では、6 成分からなる θ の 4 成分を真値に固定し、残りの 2 成分を真値付近で連続的に変化させたときの $\rho(\theta)$ の値をヒートマップとしてプロットした。変化させる 2 成分の組み合わせは、 $\{t_x, t_y\}$, $\{r_x, r_y\}$, $\{t_x, r_x\}$, $\{t_z, r_z\}$ の 4 通りを試した。

図 13-15(a) に実験で使用したカメラ画像と検出した SIFT 特徴点を、図 13-15(b)-(e) に $\{t_x, t_y\}$, $\{r_x, r_y\}$, $\{t_x, r_x\}$, $\{t_z, r_z\}$ の各成分の変化に対する評価尺度の振る舞いをそれぞれ示す。グラフの縦軸・横軸は、変化させた 2 成分の真値からのオフセット量であり、回転成分の単位はラジアン、並進成分の 0.4 は先述の通り実世界の 1m に相当する。すなわち、ヒートマップの中央 (0, 0) が真値に対応し、そこ

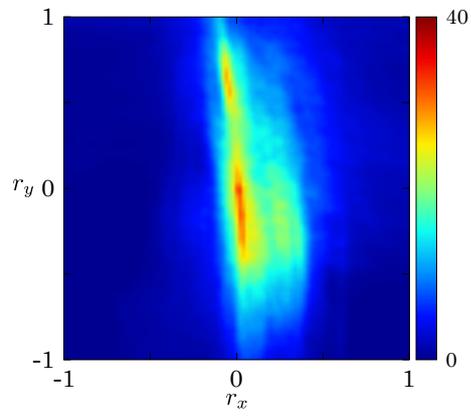
から離れるにつれて真値との誤差が増大する。多くのヒートマップにおいて、中央付近にピークがあることが見て取れる。一方、特に図14のヒートマップにおいては、真値以外にも多数のピークが存在することがわかる。これは、さまざまなスケールの特徴点あまり偏りなく画像全体に分布する場合に、データベース中の3次元点を画像上のどこに投影しても評価尺度の値が高くなるためである。また、提案する評価尺度は、特にカメラを左右前後に並進させる動きに対してピークが広がる傾向がある。この原因として、本実験で用いるデータベース中の3次元点は水平方向に広く分布しており、さらにカメラから遠い点に関しては並進に対して投影点の位置が大きく変わらないためであると考えられる。この結果から、勾配法による評価尺度の最大化において、カメラ位置姿勢の初期値が真値に十分に近い場合、提案手法はカメラ位置姿勢推定に有用であると考えられる。



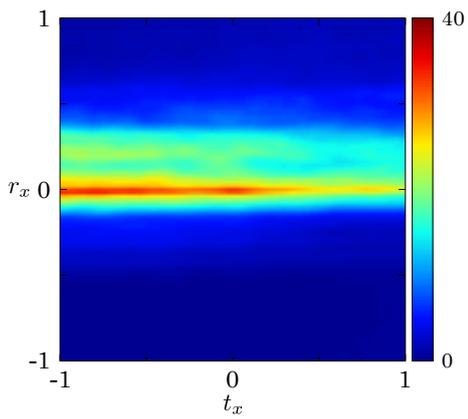
(a)



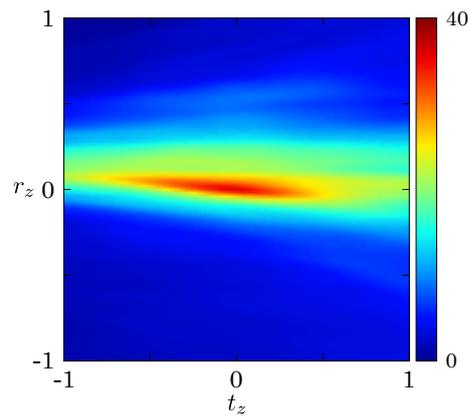
(b)



(c)

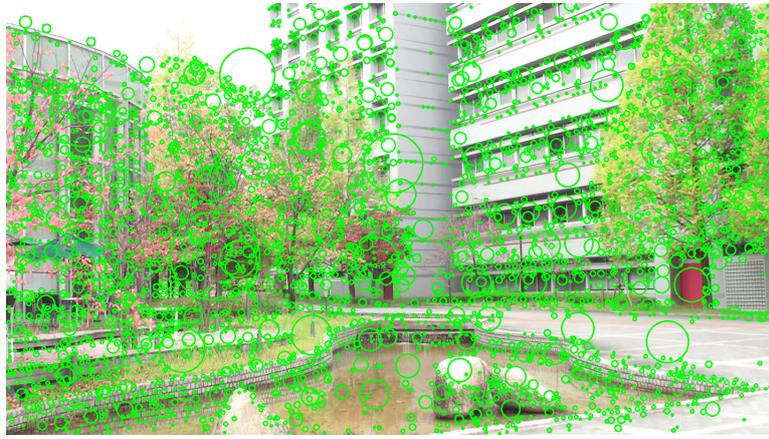


(d)

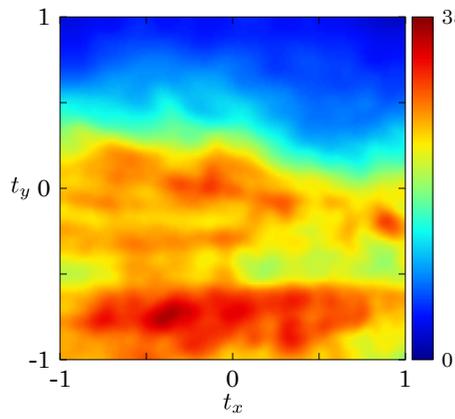


(e)

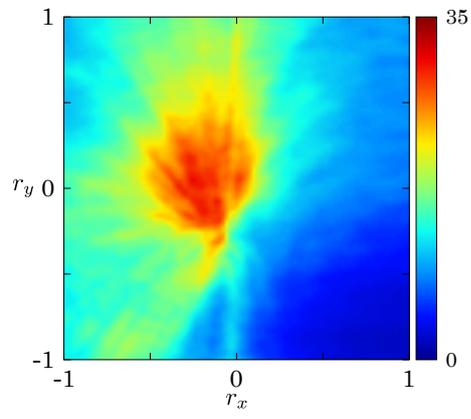
図 13: 図 12(a) のカメラ画像に対する評価尺度の振る舞い



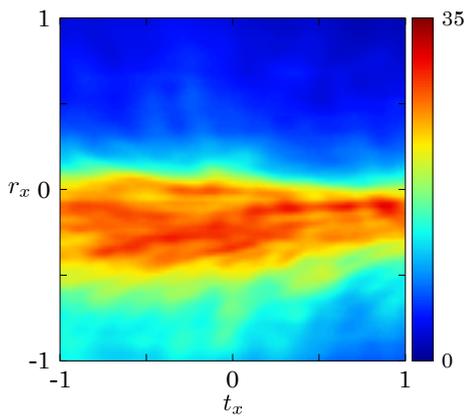
(a)



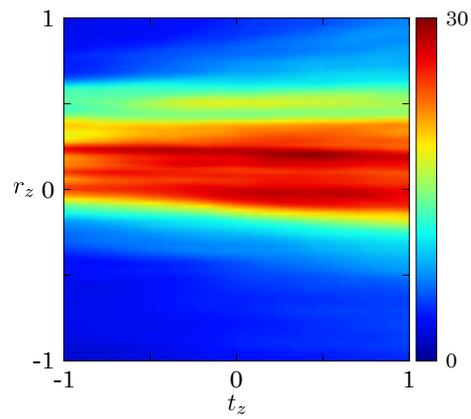
(b)



(c)

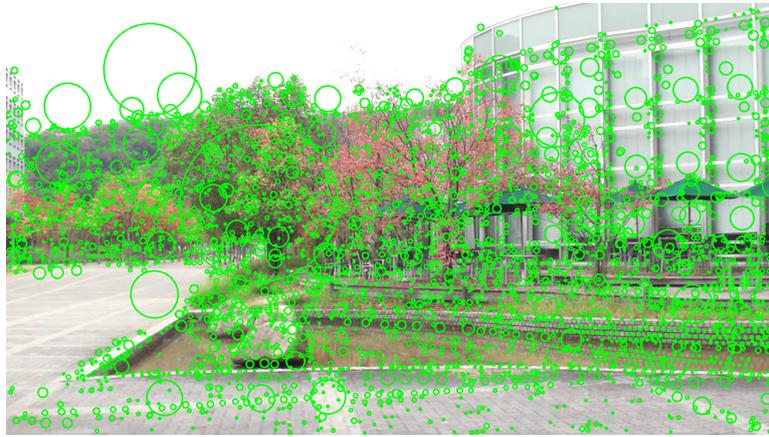


(d)

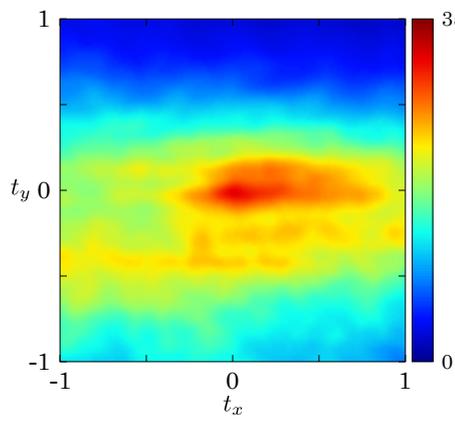


(e)

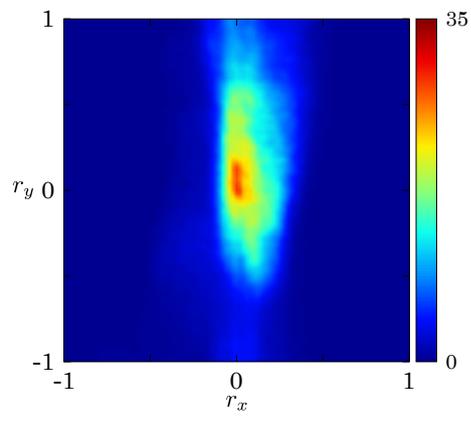
図 14: 図 12(b) のカメラ画像に対する評価尺度の振る舞い



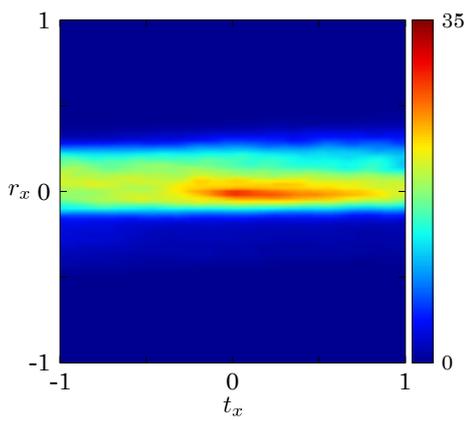
(a)



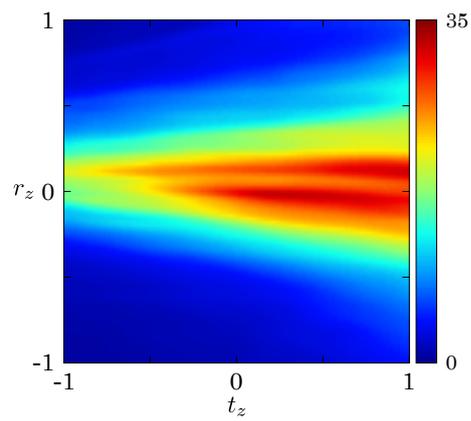
(b)



(c)



(d)



(e)

図 15: 図 12(c) のカメラ画像に対する評価尺度の振る舞い

4.3 カメラ位置姿勢推定の定量的評価

提案手法によるカメラ位置姿勢推定の精度・頑健性・計算コストを確認する実験を行った。ここでは、カメラ位置姿勢の真値に対して、式(13)に示す正規分布 $\mathcal{N}(\mathbf{0}, v^2 \mathbf{I}_3)$ に従う乱数を誤差とした与えたものを初期値として評価尺度 $\rho(\boldsymbol{\theta})$ の最大化を行った場合における回転成分と並進成分の標準偏差 $v = v_r$, $v = v_t$ (すなわち誤差の大きさ) に応じた推定精度の変化をグラフにまとめた。

$$\mathcal{N}(\mathbf{x}|\mathbf{0}, v^2 \mathbf{I}_3) = \frac{1}{(\sqrt{2\pi})^3 \sqrt{|v^2 \mathbf{I}_3|}} \exp\left(-\frac{1}{2} \mathbf{x}^\top (v^2 \mathbf{I}_3)^{-1} \mathbf{x}\right) \quad (13)$$

なお、本研究では、SFM法により3次元点群データベースを構築しており、前述の通り、得られた3次元点群の座標系の並進成分の単位は一般に実世界の単位と一致しない。そこで、本実験では、推定誤差を表す指標として3次元点の再投影誤差を用いる。ここで、カメラ位置姿勢の真値を $\boldsymbol{\theta}_{\text{GT}}$ として、これを利用して3次元点群を画像上に投影したときに画像内に含まれる3次元点の集合を \mathcal{G} とすると、推定されたカメラ位置姿勢 $\boldsymbol{\theta}$ の再投影誤差は次式により定義される。

$$E = \frac{1}{|\mathcal{G}|} \sum_{\mathbf{p} \in \mathcal{G}} \|\text{proj}_{\boldsymbol{\theta}_{\text{GT}}}^{\mathbf{p}}(\mathbf{p}) - \text{proj}_{\boldsymbol{\theta}}^{\mathbf{p}}(\mathbf{p})\| \quad (14)$$

また、 (v_r, v_t) は、 $(1.00^\circ, 0.100 \text{ m})$, $(1.41^\circ, 0.141 \text{ m})$, $(2.00^\circ, 0.200 \text{ m})$, $(2.83^\circ, 0.283 \text{ m})$, $(4.00^\circ, 0.400 \text{ m})$, $(5.66^\circ, 0.566 \text{ m})$, $(8.00^\circ, 0.800 \text{ m})$ の7通りを設定し、各組において異なる正規乱数を初期値として100回ずつ提案手法によるカメラ位置姿勢推定を行った。

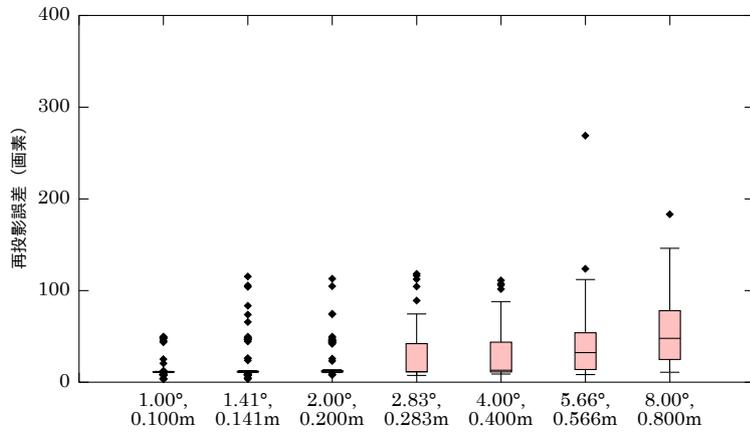
図16(a)-(c)に図12(a)-(b)のカメラ画像に対する実験結果を箱ひげ図としてそれぞれ示す。グラフの横軸は標準偏差の組、縦軸は再投影誤差であり、赤い箱の内側の線は中央値を、箱の下端・上端は第1四分位点・第3四分位点を表す。箱から伸びたひげは最小値・最大値までの区間を表しており、ひげの外側の点は外れ値を表す。この結果から、図12(a)に示すカメラ画像では、 $(v_r, v_t) = (2.83^\circ, 0.283 \text{ m})$ 以降で推定誤差が増加し、ばらつきが大きくなるものの、それ以前では安定して高い精度の推定が可能であることがわかる。一方で、図12(b)のカメラ画像では、 $(v_r, v_t) = (1.00^\circ, 0.100 \text{ m})$ においても高い精度での推定ができなかつ

た。これは、カメラ画像の特徴点が画像全体に分布しており、偏りがなかったためであると考えられる。前節の図 14 に示す結果も、このカメラ画像に対して真値付近に多くのピークを持つことから、図 12(b) の結果を支持するものである。図 12(c) のカメラ画像については、 $(v_r, v_t) = (2.00^\circ, 0.200 \text{ m})$ までは中央値が 0 付近にあることから、精度の高い推定ができた試行が半数は存在するものの、大きな誤差を含む試行も多く、安定した推定ができていない。この結果から、このカメラ画像については真値付近に別のピークが存在し、初期値によって真値もしくは他のピークのいずれかに到達したためであると考えられる。このため、提案手法を適用するには、精度のよい初期値に加え、特徴点の分布に偏りが必要であるといえる。

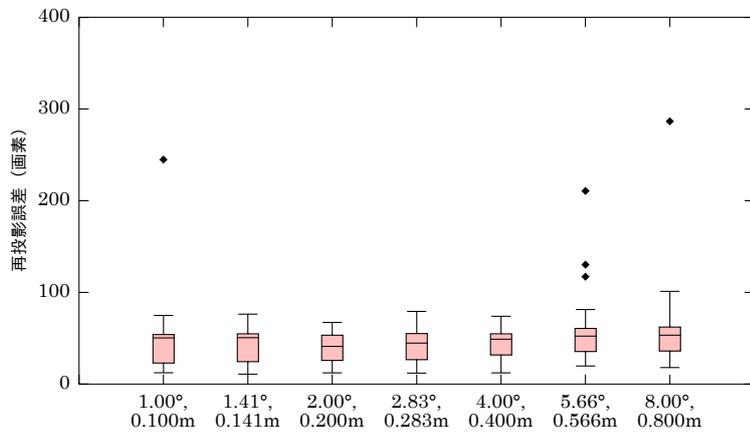
図 17-18 に、図 16(a) に示した実験結果から抜粋した成功例と失敗例におけるカメラ位置姿勢の初期値・推定値・真値での投影点を示す。左側の図は初期値 (青) と真値 (赤) での投影点を、右側の図は推定値 (青) と真値 (赤) での投影点をそれぞれ表す。成功例では、提案手法による推定を経て、投影点の真値との誤差が減少していることが見て取れる。一方の失敗例では、推定後も真値との誤差は大きいままに留まっている。ここで、図 19 に示すルックアップテーブルを見ると、右下部に大きな値が集中していることがわかる。図 18 の失敗例では、提案手法が投影点と特徴点の高密度の箇所を合わせようとし、その箇所に多数の投影点が集まったことで評価尺度 $\rho(\theta)$ が大きくなったために正しい推定ができなかったと考えられる。

計算コストの面では、1 回の推定に数秒を要しており、改善が必要であると分かった。これは、1 回の評価尺度の計算に 3 次元点群データベースのすべての点を投影する必要があること、および勾配法の効率の悪さに起因すると考える。提案手法では SFM 法を用いて得られたすべての 3 次元点を利用しているが、図 (5) を見てわかるように、点群データベース中には 3 次元点が過度に集中している箇所があり、カメラ位置姿勢推定に貢献しない 3 次元点が多く含まれていると考えられる。これらの 3 次元点を排除することで点群の規模が 10 分の 1 になった場合、10 倍の高速化が可能である。また、八分木などを用いて 3 次元点群を分割することにより、点群の投影に際して枝刈りが可能となる。例えば、画角 60 度のカメラの前後左右に点群が偏りなく分布しているとすれば、投影が必要な 3 次元点は枝刈

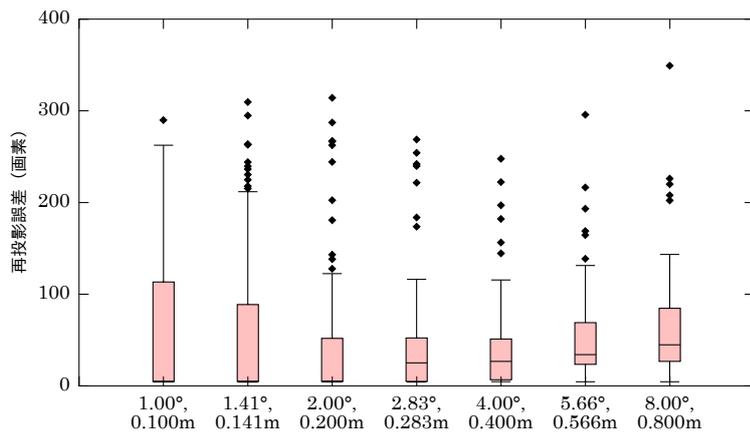
りによって6分の1程度に削減される。これら点群の削減と投影の枝刈りを合わせるにより60倍程度の高速化が見込める。加えて、最急上昇法を Levenberg-Marquardt 法 [36] などのより収束効率のよい他の勾配法に置き換えることで、さらなる高速化が可能であると考え。以上に挙げた改善を施すことにより、提案手法はリアルタイム性の要求される拡張現実感に適用できると考える。



(a) 図 12(a) のカメラ画像に対するカメラ位置姿勢の推定結果



(b) 図 12(b) のカメラ画像に対するカメラ位置姿勢の推定結果



(c) 図 12(c) のカメラ画像に対するカメラ位置姿勢の推定結果

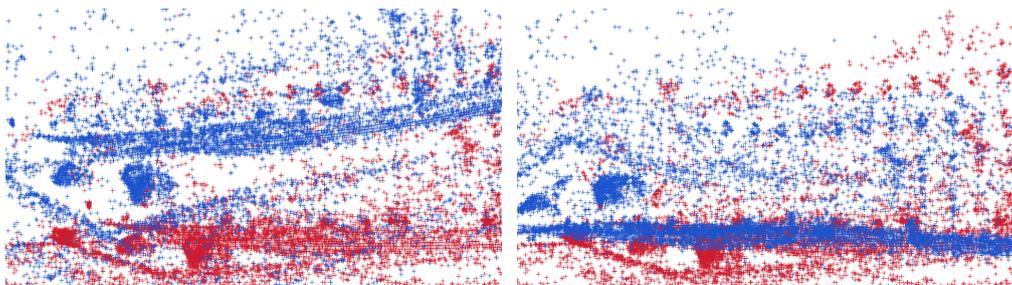
図 16: 提案手法によるカメラ位置姿勢の推定結果



(a) 初期値と真値における投影点

(b) 推定値と真値における投影点

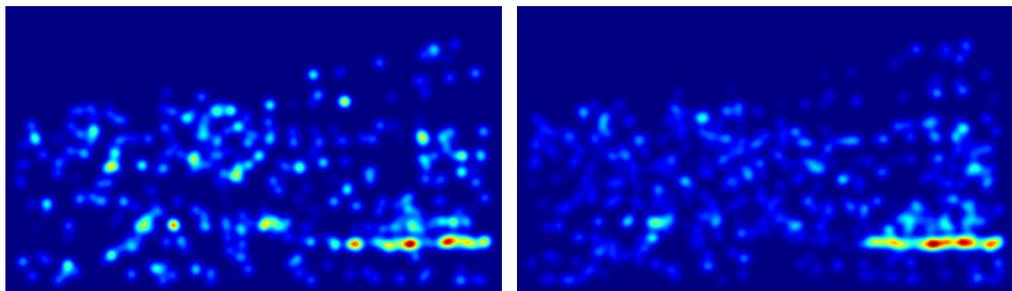
図 17: 図 12(a) のカメラ画像に対するカメラ位置姿勢の初期値・推定値・真値における投影点の比較 (成功例)



(a) 初期値と真値における投影点

(b) 推定値と真値における投影点

図 18: 図 12(a) のカメラ画像に対するカメラ位置姿勢の初期値・推定値・真値における投影点の比較 (失敗例)



(a)

(b)

図 19: 図 12(a) のカメラ画像に対するルックアップテーブルの一部

5. まとめ

本論文では、モバイル型拡張現実感への適用に向けて、事前に構築した3次元点群データベースを事前知識として用いるカメラ位置姿勢推定手法を基礎とし、記述子の抽出と記述子による特徴点の対応付けの計算コストの削減および記述子を格納するデータベースのサイズの削減のために、記述子による特徴点の明示的な対応付けを伴わないカメラ位置姿勢手法を提案した。提案手法は、あるカメラ位置姿勢のもとで3次元点群を投影して得られる投影点と、カメラで撮影した画像から検出した特徴点の、画像平面上での分布の一致度合いをカメラ位置姿勢の評価尺度とし、これを勾配法によって最大化することでカメラ位置姿勢を推定した。実験では、提案したカメラ位置姿勢の評価尺度の振る舞いと、提案手法によるカメラ位置姿勢推定の精度を確認した。実験の結果、提案した評価尺度は真値においてピーク値を取ることが確認でき、投影点と特徴点の分布の一致度合いがカメラ位置姿勢を評価する尺度として有用であることを示した。一方で、提案手法が高い精度でカメラ位置姿勢を推定するためには、十分によい初期値と特徴点の分布の偏りが必要であることが明らかとなった。以上の結果から、提案手法は、GPSや地磁気センサなどを持つモバイル端末上での利用や、時間的な連続性が期待できるカメラ位置姿勢のトラッキングなどでの応用が考えられる。データベースサイズに関しては、4.1節で述べた通り、提案手法で構築した22,147点のデータベースのサイズが346 kBであった。仮に、従来手法のように、点群データベースの22,147点すべてに128次元のSIFT記述子を付与することを想定すると、SIFT記述子は128 Byteで表現されることから、各点に1つの記述子のみを付与するとしても、さらに約2.7 MBの容量が必要になる。従来手法の多くは、点の対応付けの精度を高めるために各3次元点に複数の記述子を付与しているため、実際にはさらに多くの容量を必要とする。一例として、Liら [4] が公開している点群データベース「Dubrovnik6K」は、全2,106,456点、容量3.68 GBであり、1点あたり1,876.54 Byteの記憶容量を消費している。それに対して本手法のデータベースは1点あたり16.00 Byteであり、Dubrovnik6Kに比べ100倍以上のストレージ効率がある。

今後の展望として、評価尺度の改良と計算の高速化が挙げられる。本論文で提

案した点の位置とスケールを利用した評価尺度は、点の色やオリエンテーションも導入できると考える。色を導入する場合は、例えば離散化した色空間の各々に対応するルックアップテーブルを用いる方法が考えられる。ただし、色は照明条件や季節の変化の影響を受けやすいため、導入にあたってはそうした問題を考慮する必要がある。オリエンテーションを導入する場合は、画像平面上での向きとして算出されるオリエンテーションの3次元空間中での扱いについて考慮する必要がある。評価尺度の高速化については、提案手法では3次元点群のすべての点を投影する必要があるが、点群の削減や投影の枝刈りにより計算コストを削減できると考える。それに加えて、最急上昇法よりも収束効率のよい他の勾配法を採用することで、提案手法の高速化が実現できると考える。

謝辞

本研究を進めるにあたり，懇切なる御指導，御鞭撻を賜りました視覚情報メディア研究室 横矢直和 教授に心から感謝致します。また，本研究を通して，有益な御助言，御鞭撻を頂いたインタラクティブメディア設計学研究室 加藤博一 教授に厚く御礼申し上げます。そして，本研究のあらゆる面において，始終暖かく細やかな御指導をして頂いた視覚情報メディア研究室 佐藤智和 准教授に深く感謝致します。佐藤智和 准教授には，本研究のテーマ設定から日々のミーティング，本論文の執筆・添削，発表の練習に至るまでの的確な御指導を頂きました。さらに，本研究への御指導，御助言をして頂いた視覚情報メディア研究室 中島悠太 助教，河合紀彦 助教に深く感謝致します。中島悠太 助教には，常日頃から本研究に関する多大な御意見と親身な御指導を頂きました。また，研究室での生活を支えてくださった視覚情報メディア研究室 石谷由美 女史に感謝致します。最後に，研究活動だけでなく日々の生活においても大変お世話になった視覚情報メディア研究室諸氏に心より感謝致します。

参考文献

- [1] Kato, H., Billinghurst, M. “Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System”. In *Proc. IEEE and ACM Int. Workshop on Augmented Reality*, pp. 85–94. 1999.
- [2] Kato, H., Billinghurst, M., Poupyrev, I., Imamoto, K., Tachibana, K. “Virtual Object Manipulation on a Table-top AR Environment”. In *Proc. IEEE and ACM Int. Symp. Augmented Reality*, pp. 111–119. 2000.
- [3] Irschara, A., Zach, C., Frahm, J.-M., Bischof, H. “From Structure-from-motion Point Clouds to Fast Location Recognition”. In *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, pp. 2599–2606. 2009.
- [4] Li, Y., Snavely, N., Huttenlocher, D. “Location Recognition using Prioritized Feature Matching”. In *Proc. European Conf. Computer Vision: Part II*, pp. 791–804. 2010.
- [5] Li, Y., Snavely, N., Huttenlocher, D., Fua, P. “Worldwide Pose Estimation using 3D Point Clouds”. In *Proc. European Conf. Computer Vision: Part I*, pp. 15–29. 2012.
- [6] Sattler, T., Leibe, B., Kobbelt, L. “Fast Image-based Localization using Direct 2D-to-3D Matching”. In *Proc. IEEE Int. Conf. Computer Vision*, pp. 667–674. 2011.
- [7] Horaud, R., Conio, B., Le Boulleux, O., Lacolle, B. “An Analytic Solution for the Perspective 4-point Problem”. In *IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, pp. 500–507. 1989.
- [8] Gao, X.-S., Hou, X.-R., Tang, J., Cheng, H.-F. “Complete Solution Classification for the Perspective-three-point Problem”. *IEEE Trans. Pattern Analysis and Machine Intelligence*, volume 25, no. 8, pp. 930–943, 2003.

- [9] Lepetit, V., Moreno-Noguer, F., Fua, P. “EPnP: An Accurate $O(n)$ Solution to the PnP Problem”. *Int. J. Computer Vision*, volume 81, no. 2, pp. 155–166, 2009.
- [10] Newman, J., Ingram, D., Hopper, A. “Augmented Reality in a Wide Area Sentient Environment”. In *Proc. IEEE and ACM Int. Symp. Augmented Reality*, pp. 77–86. 2001.
- [11] Tenmoku, R., Kanbara, M., Yokoya, N. “A Wearable Augmented Reality System for Navigation using Positioning Infrastructures and a Pedometer”. In *Proc. IEEE and ACM Int. Symp. Mixed and Augmented Reality*, pp. 344–345. 2003.
- [12] Feiner, S., MacIntyre, B., Hollerer, T., Webster, A. “A Touring Machine: Prototyping 3D Mobile Augmented Reality Systems for Exploring the Urban Environment”. In *Proc. IEEE Int. Symp. Wearable Computers*, pp. 74–81. 1997.
- [13] Gleue, T., Dähne, P. “Design and Implementation of a Mobile Device for Outdoor Augmented Reality in the Archeoguide Project”. In *Proc. Conf. Virtual Reality, Archeology, and Cultural Heritage*, pp. 161–168. 2001.
- [14] Höllerer, T., Feiner, S., Pavlik, J. “Situated Documentaries: Embedding Multimedia Presentations in the Real World”. In *Proc. IEEE Int. Symp. Wearable Computers*, pp. 79–86. 1999.
- [15] Piekarski, W., Hepworth, D., Demczuk, V., Thomas, B., Gunther, B. “A Mobile Augmented Reality User Interface for Terrestrial Navigation”. In *Proc. Australasian Computer Science Conf.*, pp. 122–133. 1999.
- [16] Vlahakis, V., Karigiannis, J., Tsotros, M., Gounaris, M., Almeida, L., Stricker, D., Gleue, T., Christou, I. T., Carlucci, R., Ioannidis, N. “Archeoguide: First Results of an Augmented Reality, Mobile Computing

- System in Cultural Heritage Sites”. In *Proc. Conf. Virtual Reality, Archeology, and Cultural Heritage*, pp. 131–140. 2001.
- [17] Thrun, S., Bennewitz, M., Burgard, W., Cremers, A. B., Dellaert, F., Fox, D., Hahnel, D., Rosenberg, C., Roy, N., Schulte, J., Schulz, D. “MINERVA: A Second-generation Museum Tour-guide Robot”. In *Proc. IEEE Int. Conf. Robotics and Automation*, volume 3, pp. 1999–2005. 1999.
- [18] Thrun, S., Burgard, W., Fox, D. “A Real-time Algorithm for Mobile Robot Mapping with Applications to Multi-robot and 3D Mapping”. In *Proc. IEEE Int. Conf. Robotics and Automation*, volume 1, pp. 321–328. 2000.
- [19] Castellanos, J. A., Neira, J., Tardos, J. D. “Multisensor Fusion for Simultaneous Localization and Map Building”. *IEEE Trans. Robotics and Automation*, volume 17, no. 6, pp. 908–914, 2001.
- [20] Se, S., Lowe, D. G., Little, J. J. “Vision-based Global Localization and Mapping for Mobile Robots”. *IEEE Trans. Robotics*, volume 21, no. 3, pp. 364–375, 2005.
- [21] Davison, A. J., Reid, I. D., Molton, N. D., Stasse, O. “MonoSLAM: Real-time Single Camera SLAM”. *IEEE Trans. Pattern Analysis and Machine Intelligence*, volume 29, no. 6, pp. 1052–1067, 2007.
- [22] Klein, G., Murray, D. “Parallel Tracking and Mapping for Small AR Workspaces”. In *Proc. IEEE and ACM Int. Symp. Mixed and Augmented Reality*, pp. 1–10. 2007.
- [23] Klein, G., Murray, D. “Parallel Tracking and Mapping on a Camera Phone”. In *Proc. IEEE Int. Symp. Mixed and Augmented Reality*, pp. 83–86. 2009.
- [24] Castle, R., Klein, G., Murray, D. “Video-rate Localization in Multiple Maps for Wearable Augmented Reality”. In *Proc. IEEE Int. Symp. Wearable Computers*, pp. 15–22. 2008.

- [25] Cummins, M., Newman, P. “FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance”. *Int. J. Robotics Research*, volume 27, no. 6, pp. 647–665, 2008.
- [26] Drummond, T., Cipolla, R. “Real-Time Visual Tracking of Complex Structures”. *IEEE Trans. Pattern Analysis and Machine Intelligence*, volume 24, no. 7, pp. 932–946, 2002.
- [27] Taketomi, T., Sato, T., , Yokoya, N. “Real-time and Accurate Extrinsic Camera Parameter Estimation using Feature Landmark Database for Augmented Reality”. *Int. J. Computers and Graphics*, volume 35, no. 4, pp. 768–777, 2011.
- [28] Snavely, N., Seitz, S. M., Szeliski, R. “Photo Tourism: Exploring Photo Collections in 3D”. *ACM Trans. Graphics*, volume 25, no. 3, pp. 835–846, 2006.
- [29] Snavely, N., Seitz, S., Szeliski, R. “Skeletal Graphs for Efficient Structure from Motion”. In *IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, pp. 1–8. 2008.
- [30] Snavely, N., Seitz, S. M., Szeliski, R. “Modeling the World from Internet Photo Collections”. *Int. J. Computer Vision*, volume 80, no. 2, pp. 189–210, 2008.
- [31] Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S. M., Szeliski, R. “Building Rome in a Day”. *Communications of the ACM*, volume 54, no. 10, pp. 105–112, 2011.
- [32] Lowe, D. G. “Distinctive Image Features from Scale-Invariant Keypoints”. *Int. J. Computer Vision*, volume 60, no. 2, pp. 91–110, 2004.
- [33] Bay, H., Ess, A., Tuytelaars, T., Van Gool, L. “Speeded-up Robust Features (SURF)”. *Computer Vision and Image Understanding*, volume 110, no. 3,

pp. 346–359, 2008.

- [34] Wu, C. “VisualSFM: A Visual Structure from Motion System”. URL <http://ccwu.me/vsfm/>.
- [35] Wu, C. “SiftGPU: A GPU Implementation of Scale Invariant Feature Transform (SIFT)”. URL <http://cs.unc.edu/ccwu/siftgpu/>.
- [36] Marquardt, D. W. “An Algorithm for Least-squares Estimation of Nonlinear Parameters”. *J. Society for Industrial and Applied Mathematics*, volume 11, no. 2, pp. 431–441, 1963.