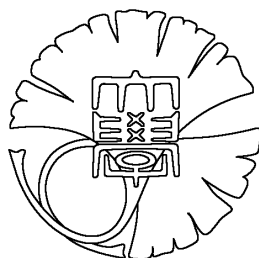


修士学位論文

気球搭載型レンジセンサから得られた距離画像の
位置姿勢及び運動パラメータの同時推定



平成 15 年度

東京大学学際情報学府

学際情報学専攻実践情報学コース

学籍番号 26217

廣田 祐一郎

指導教官 池内 克史 教授

論文要旨

気球搭載型レンジセンサーから得られた距離画像 の位置姿勢及び運動パラメータの同時推定

実践情報学コース

学籍番号 26217

廣田 祐一郎

近年、高精度なレーザーレンジセンサーから得られる3次元の計測データ(距離画像)を用いて、実物体をサイバー空間上で再現する研究が行われている。我々は主に大仏、寺院等の大規模な文化財を効率的にモデリングする手法の開発を行ってきた。通常の3次元形状の計測は、レーザーレンジセンサーを地上に設置し、対象物体を余すことなく測定できるよう複数の方向からスキャンを行う。しかし、計測する対象物とそれを取り巻く環境によっては、例えば寺院の屋根のように、地上からでは計測できない箇所が生じてしまう。本論文では地上から計測不可能な領域を、空中から効率的に計測する気球搭載型センサー(Flying Laser Range Finder, FLRF)を提案する。さらに、このセンサーを用いて計測を行う場合、気球の性質上、センサーの位置が風等の環境により変動し画像が歪むといった問題が発生する。そこで本論文では気球センサーによって得られた距離画像の歪みを、センサーの運動パラメータを推定することで補正する、気球センサーの位置姿勢と運動パラメータの同時推定レジストレーション手法を提案する。

謝辞

この論文を執筆するにあたって研究室内外の方に大変お世話になりました。

私に池内研究室という活気に満ち溢れたすばらしい環境を与えて頂き、本研究をご指導してくださった池内克史教授に深く感謝の意を表します。日頃より研究に関してご指導いただき、時には朝まで議論に付き合っていたいただいた山崎さん、高松さん、大石さん。飲み込みが遅かったり、頷いておいてなんにもやらなかったりしてあきれさせてしまったこともあるかもしれませんが私が池内研究室で得たものの多くは先輩諸氏の助言・議論の中で生まれたものだと考えております。本当にありがとうございました。大石さんには飲み会の二次会以降でもお世話になりました。感謝しています。また研究を共に進めてくださった増田さん、長谷川さんには心から感謝しています。増田さんは特に研究初期に、私がしつこいほど質問をしたときも快く答えてくれました。また、個人的な計測にもご協力頂きました。深く感謝しております。長谷川さんには気球センサでお世話になりました。実験の手配等まかせっきりになってしまいました。ありがとうございました。また、気球センサについては小川原さんにお世話になりました。研究に関するもお世話になりました。お盆期間中に突然ロボ部屋に行っても、小川原さんはいつも通り椅子に座っていて、私のために時間を割いてくれました。ありがとうございました。生研での生活は平原さんにお世話になりました。初めて研究室に泊まったときシャンプーのみならずバスタオルまで貸してくれたことは今でも記憶に焼きついています。ありがとうございました。

この研究におけるデータ計測では多くの方々のお世話になりました。カンボジアのデータ計測では特に中澤さんにお世話になりました。計測のスケジュール管理、現地での交渉等何もかもをお世話になっていながら勝手にどこかに行ってお迷惑をおかけしました。この場を借りて改めてお詫びすると共にお礼申し上げます。気球センサでの計測は池内研究室のスタッフだけでなく、現地のスタッフの方々にもお世話になりました。なかでも二人の運転手さんにはお世話になりました。ありがとうございました。

最後に私の研究生生活を精神的にも、経済的にも支えてくださった両親に感謝いたします。



廣田 祐一郎

目次

1	序論	1
1.1	研究背景	1
1.2	研究目的	1
1.3	論文概要	2
2	実物体のモデル化	3
2.1	デジタル保存	3
2.2	デジタル復元	5
3	気球を用いた空中モデリングシステム概要	11
4	ハードウェア	14
4.1	関連研究	14
4.2	気球搭載型センサ	14
5	ソフトウェア	21
5.1	関連研究	21
5.2	位置姿勢と運動パラメータの同時推定手法	24
5.3	実装	25
5.4	精度の向上についての考察	27
5.4.1	評価式の改良	27
5.4.2	簡略化	33
6	ソフトウェアの評価	34
6.1	実データにおける実験	34
6.2	精度の評価	34
6.2.1	点の間引きの影響評価	36
6.2.2	同時推定レジストレーションの精度評価	40

6.2.3	改良アルゴリズムの比較実験	42
7	まとめ	45

第 1 章

序論

1.1 研究背景

貴重な文化財の状態を記録し保存することは世界各地で必要とされている。遙か古代に造られた建造物の多くは石や岩を加工して造られたものである。こうした建造物は長い年月風雨にさらされ年々劣化が進行している。建物の表面は崩れ、建物の一部を残して倒壊してしまった例も多い。そして現在残っている建造物も時間の経過と共に劣化し続けている。また水害や地震等の災害によって文化財が失われてしまうことも多い。特に火山国の日本では地震による建物の倒壊が後を絶たない。また日本の建築物は歴史的にほとんどが木造だったので、落雷や失火による火災によって文化財が焼失してしまうこともしばしばであった。さらにそれらに加えて人為的な理由によって貴重な文化財が失われることも多い。その中でも戦争の被害を受けることが最も多い理由の一つとして挙げられるが、その原因ともなる宗教的な問題はいつの時代も避けられないものである。現在最も優先されなければならない課題はこのように危険にさらされている文化財の保護である。しかし万が一このような文化財が破損し、あるいは完全に失われてしまった場合は、それを修復、復元しなければならない。この場合、古くから存在する文化財の多くは建造当時の図面や修復の手掛かりとなる資料が残されている事が少ないため、修復や復元は非常に困難な作業となる。そこで現存する貴重な文化財の状態を測定し、これを記録・保存する手法が必要とされているのである。

1.2 研究目的

我々はこれまでに奈良、鎌倉大仏そしてカンボジアのバイヨン寺院といったような大規模文化財をモデリングしてきた [1]。基本的にこれらの物体を計測するときは地上の安定した地点からセンサを三脚上に取り付けて行ってきた。しかし、この方法では対象物が大規模であるために、地上から見えない屋根の頂等でしばしばオクルージョンが発生してし

まうという問題があった．そしてそのような場合は地上から計測できない部分をセンサが計測できるように足場を組んで対処してきた．これには多大なコストを要するだけでなく設置スペースの問題から入り組んだ寺院では足場を組むことさえできないといった問題があった．そこで本研究では既存の手法では困難であった入り組んだ寺院等で，地上から計測できない部分を効率よく計測するために気球から吊り下げられた気球搭載型センサを用いた空中モデリングシステムを開発することを目的とした．

1.3 論文概要

今回提案するシステムの研究開発は大きく分けて2つのフェーズから構成される．一つ目は気球搭載型センサの製作，すなわちハードウェア開発である．完成した気球センサから得られた距離画像は，気球特有の揺れによりデータに歪みを持つ．このデータを実際にモデリングに用いるには歪みを補正する必要がある．よってシステムを構成する二つ目のフェーズとしてレジストレーションによる歪み補正(気球の運動パラメータ推定)，すなわちソフトウェアの開発を行った．

本論文の構成について述べる．第2章では本研究の基盤となる研究である実物体のモデル化についてその概要を述べる．第3章では本論文で提案する気球からの空中モデリングシステムの概要について述べる．第4章では気球センサの開発，すなわちシステムにおけるハードウェアに関する領域について述べる．第5章ではセンサから得られた画像からセンサの位置姿勢・運動パラメータを同時推定するレジストレーション手法，すなわちシステムにおけるソフトウェアに関する領域について述べ，さらにこの手法のパラメータ推定精度を実験・検証する．最後に第7章で本研究の総括をする．

第 2 章

実物体のモデル化

本論文で本題とする気球上からの空中モデリングシステムは実物体のモデル化 (Modeling from Reality) の一環と位置づけられる。本章では論文の本題に先立って我々がこれまで行ってきた実物体のデジタル保存とそれをサイバー空間で再現するデジタル復元の概要について述べる。

2.1 デジタル保存

失われていく文化財を記録し保存するためには現実物体から 3 次元モデルを自動生成する実物体のモデル化の概念によって提案されている手法を用いる。コンピュータビジョンやコンピュータグラフィックスの分野では現実物体を測定し、テレビゲームや Virtual Reality 等で使用される 3 次元モデルを自動生成する研究が盛んに行われている。またその応用として美術品を 3 次元モデル化した研究もある [2]。そこでこのような手法を応用して文化財を測定、モデル化しデジタル的な保存を試みる。実物体のモデル化は主に幾何情報・光学情報の取得という大きく分けて二つの過程があり双方とも重要な要素であるが、次節以降では本研究で扱う幾何情報つまり形状のモデル化について紹介する。

距離画像 (距離データ) の測定

物体の形状をモデル化するためには、まず物体に対して複数方向から距離データを測定する必要がある。距離データとはある点から物体までの距離を表したもので、このデータの集合によって物体表面の形状を表現する事ができる。一般にレーザーレンジセンサと呼ばれる測定機器は、これらの機器が定める原点から物体表面までの距離を格子状に測定する (図 2.1) 事が可能である。この格子状に並べられた距離データは 2 次元の画像と対比させて距離画像とも呼ばれる。しかしレンジセンサは CCD 等の受光素子が光線を得られる範囲でしか形状を測定できない。つまり見えている部分しか測定できないということであ

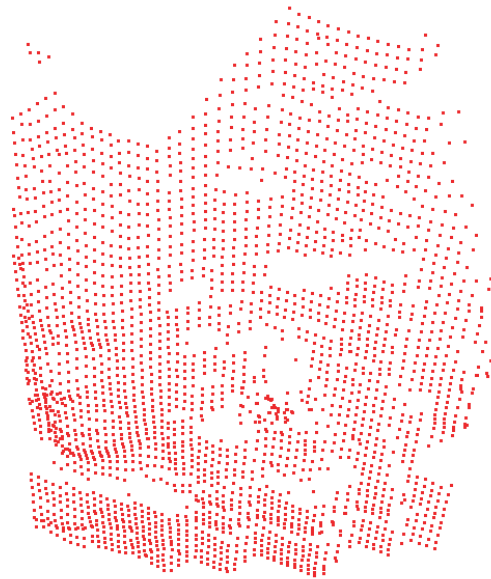


図 2.1: レンジセンサから得られる 1 枚の距離画像の 3 次元表現

る．そのため，ある物体の形状を測定するためには物体全体を覆うように複数方向から距離データを測定しなければならない．デジタル保存のための距離データ測定にはレーザーを使用したレンジセンサが適していると考えられる．非接触で実物体の測定する方法は幾つか存在するが，これらの手法は主に受動的なシステムと能動的なシステムとに分類される．受動的なシステムでは光源環境等を特定せず CCD カメラ等から得られる画像を解析することによって物体の見えや形状を測定する．このようなシステムは高速に形状の測定を行えるが，対応点探索や画像解像度の問題等からあまり高い精度は得られない．一方，能動的なシステムでは対象物体にレーザーや特殊な光源パターンを投影しその反射光の状態によって形状を測定する．受動的なシステムに比べて測定のためにある程度長い時間を必要とするが高精度で高解像度のデータを得ることが可能である．特にレーザーを使用した距離測定器（レーザーレンジセンサ）は非常に高精度な距離データを測定することができる．そのため保存を目的として建築物のように静的な物体を対象とする場合は，高精度なデータを測定可能なレーザーレンジセンサが適していると考えられる．またレーザーレンジセンサは使用するレーザーの波長や測定方式によって精度や測定範囲を選択可能なため，木造建築や仏像から小さな工芸品に至るまで多種多様な有形文化財のデジタル化に適用できる．

形状モデル生成

レーザーレンジセンサを用いて測定された複数の距離データから一つの3次元モデルを生成する。レンジセンサを用いて測定された距離データは単なる点の集合でしかない。そこで幾つかの過程を経ることによって一つの形状モデルを生成する。この生成される形状モデルは頂点とそれを結ぶ三角形の接続関係で表現されたメッシュモデルである(図2.3)。この一つのメッシュモデルを生成するためには「位置合わせ」「統合」「穴埋め」という処理が必要となる。以下に簡単にこれらの処理について説明する。

- 位置合わせ(レジストレーション)

複数方向から得られた各距離画像はレーザーレンジセンサが定めるある原点からの相対的な距離を表している。そのため各距離画像はそれぞれ異なる座標系を持っている。そこで、これらの座標系を統一する位置合わせという処理が必要である。つまり各距離画像を測定した位置と方向を推定する処理である(図2.2下)。

- 統合(マージング)

この処理では各距離画像を統一して一つのモデルを生成する。各距離画像はメッシュモデルに変換されているが、それぞれは異なったモデルである。そのため、この複数のメッシュモデルを統一して一つのモデルとしなければならない。これが統合と呼ばれる処理である(図2.2右)。

- 穴埋め

「位置合わせ」「統合」という過程を経たモデルは過不足無く統一されたモデルでなければならない。しかし、測定できなかった部分や統合の際に整合が取れず失われてしまった箇所等が存在する。そこでこの不足箇所(穴)を補間する穴埋めという処理が必要である(図2.4,2.5)。

上記のように幾つかの過程を経て物体の形状モデルは生成される。これらの手法を用いて文化財のデジタル保存を行っていく。

2.2 デジタル復元

失われた文化財の過去の姿をコンピュータグラフィックス(CG)によってデジタル的に復元する。前述のようにこれまでに多くの貴重な文化財が様々な理由によって失われている。これらの失われてしまった文化財を復元することは歴史学や建築学的に重要なことであり、また宗教的にも大きな意味を持つ。しかし実際に巨大な建造物を復元するのは容易な事ではなく、縮小された模型による復元も物理的に様々な制約を受ける等の問題点があ

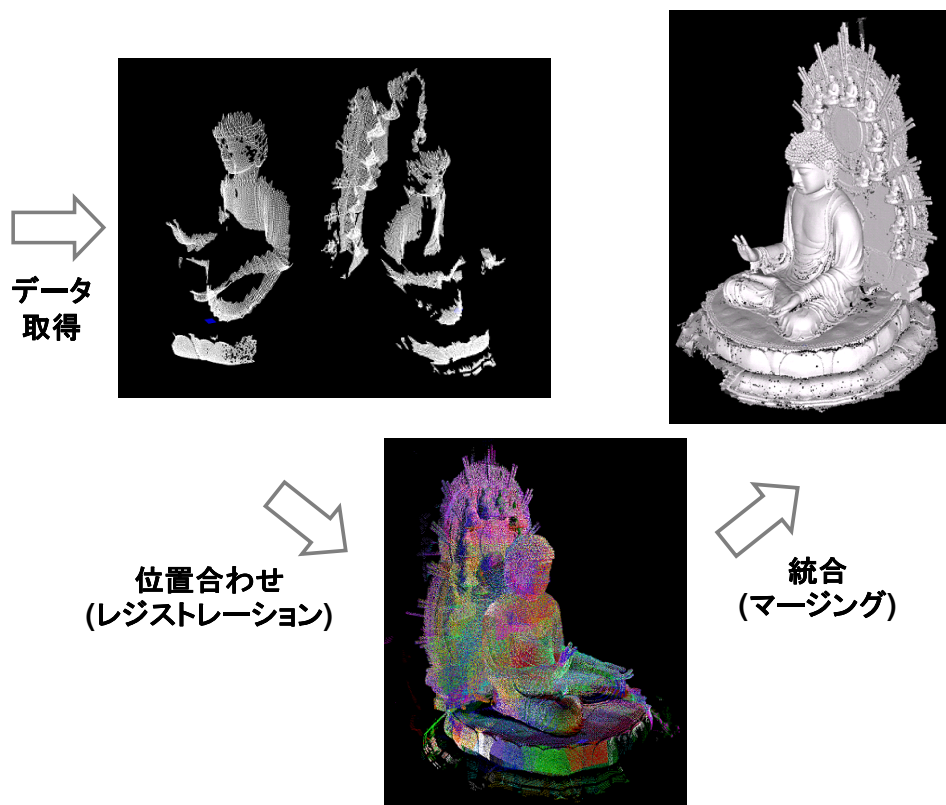


図 2.2: 実物体のモデル化

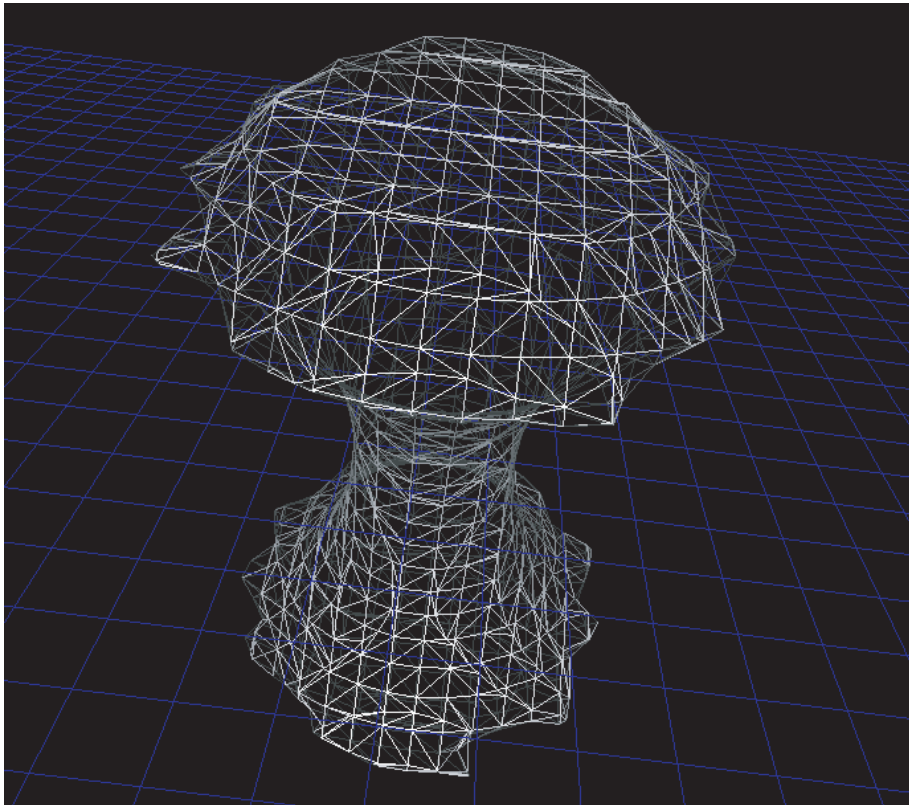


図 2.3: 簡単なメッシュモデルの例



図 2.4: 穴埋め処理前

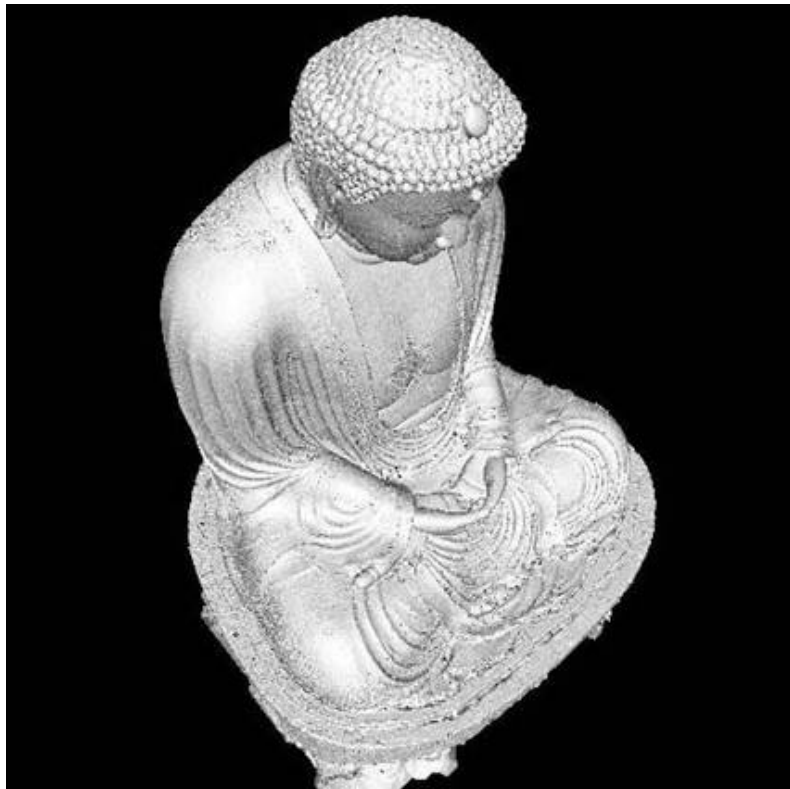


図 2.5: 穴埋め処理後

る．そこで，より多くの利点を持つと考えられるCGによるデジタル復元を試みる．CGによる復元は実物の復元や縮小模型の作成に比べればはるかに容易であるし，作成されたモデルが劣化や災害によって失われることも無い．またデジタルで作成されたモデルは端末さえあれば時間や場所に限られずに鑑賞することができる．そしてあらゆる自由な視点から対象を眺めることが可能であり，実世界では不可能な視点から眺めることも可能にする．また単純に見えを再現するだけでなくモデルに物理的な要素を持たせる事によって建築物の耐久性等のシミュレーションを行うことで，文献値の信頼性を検証することも可能である．このようにCGによるデジタル復元は多くの利点を持つので，その手法を体系化できれば今後様々な分野で大きな研究の成果につながると考えられる．

形状の推定

復元モデルを作成するためにはまず対象となる文化財の当時の様子を推定しなければならない．古い建造物は建設時の図面が残っていることは少ないが，修復時や再建時には寸法が測定され資料として伝えられていることもある．また信憑性は定かではないが様々な文献中に寸法や姿形等が部分的に記述されている場合もある．そこで，このような文献資料を元に当時の形状を推定していく．また，このとき常に文献値の信憑性も十分に検討していく必要がある．

復元モデルの作成

形状を推定した後にこれを元にして復元されたCGモデルの作成を行う．通常このような復元モデルの作成はモデラと呼ばれるCADを使って行われる．このCADを使ったCGモデルの作成は，実世界の復元に比べればはるかに低コストで短時間で可能である．しかしCADの扱いには専門の技術が必要であり，またクリエイターと呼ばれる専門の技術者でも多くの時間と労力を費やしてモデルを作成しているのである．また仏像のような曲線が多くまた複雑な形状のモデルを作成することは，専門の技術者達にとっても非常に困難な作業である．そこで予めデジタル保存されたモデルを利用して復元モデルを作成する(図2.6)．これにより簡便に短時間で複雑なモデルを作成する事が可能となる．

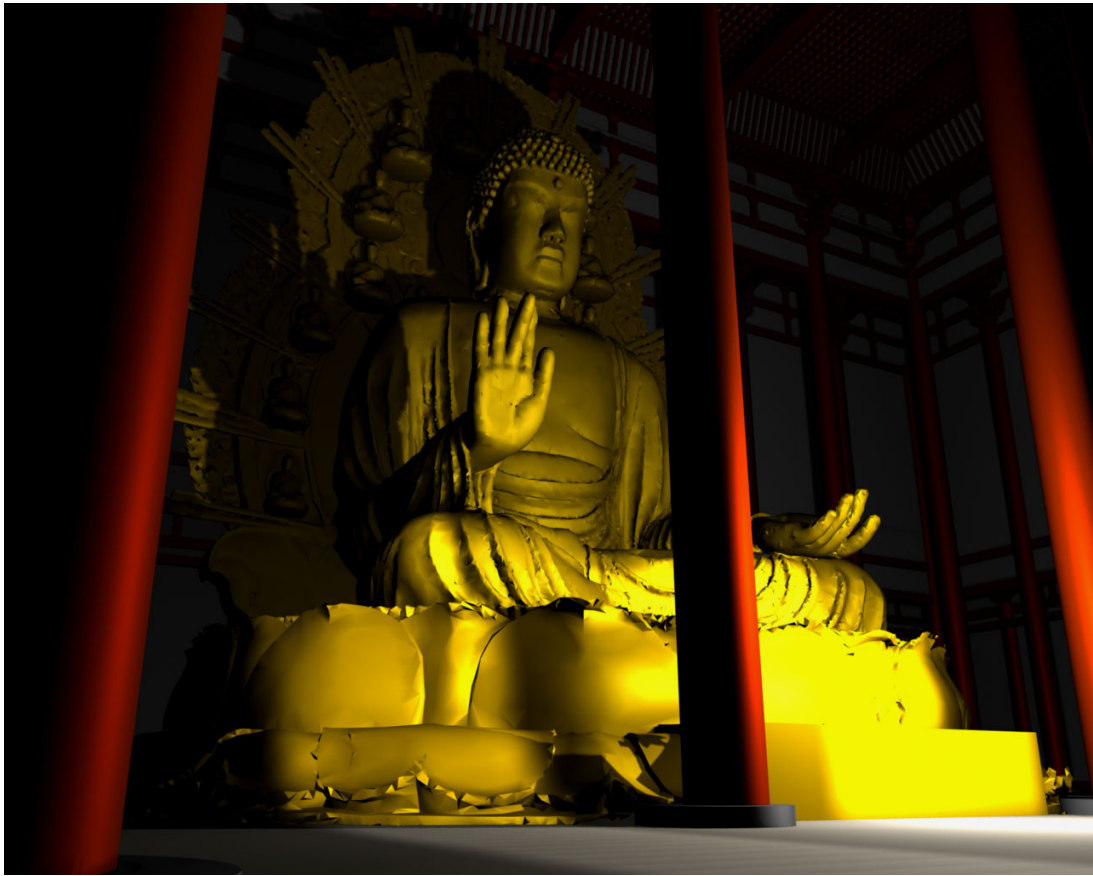


図 2.6: 推定復元した開眼当時の奈良大仏

第3章

気球を用いた空中モデリングシステム概要

我々はこれまでに奈良，鎌倉大仏そしてカンボジアのバイヨン寺院といったような大規模な文化財を対象にモデリングを行ってきた [1]．基本的にこれらの物体を計測するときは安定した地点からセンサを三脚上に取り付けて行い，地上から見えない屋根の頂等はセンサが計測できるように足場を組み，その上からセンサを向けて計測する対処してきた．しかしこの方法は多大なコストを要するだけでなく設置スペースの問題から入り組んだ寺院では足場を組むことさえできないという問題があった．

このような地上からの計測ではオクルージョンとなってしまう部分の計測方法としてはクレーン車等に吊り下げて行う方法もあるが，我々の計測対象としている寺院等は，狭い敷地に建物が入り組んでいるため，大型車両が敷地に入ることは困難なことが多い．そこで我々はこの問題の解決策として気球からの空中モデリングシステムを提案する．

本システムの流れを図 3.1 に示す．提案手法はスキャニングとレジストレーションという二つのフェーズからなる．スキャニングはまずセンサを安定して設置できる地上等から，対象物体をスキャンする．これをセンサの設置場所を何回も変えて行い，可能な限りオクルージョンがなくなるまで繰り返す．本研究ではこの地上からの文化財の計測は市販のレーザーレンジセンサ，CYRAX 2500(図 3.2，表 3.1)を用いて行った．次にこの作業を終えて，安定した場所からどうしても計測することができなかったオクルージョンについて本研究で開発した気球搭載型センサ (Flying Laser Range Finder, FLRF) を用いて空中から計測する．

スキャニングにより十分な数の距離画像を得たら，次はこれらのデータをレジストレーションして一つの3次元モデルにする．本手法では気球搭載型センサで計測したデータレジストレーションは気球センサにより得られた歪んだ画像 (モデル画像) と，それと十分な重なり部分を持つ CYRAX 2500 により得られた正確な画像 (シーン画像) のペアで行う．このレジストレーション過程でモデル画像はシーン画像に対して位置合わせされると同時に歪み補正も行われる．つまり本提案手法はモデル画像の位置姿勢パラメータと気球セン

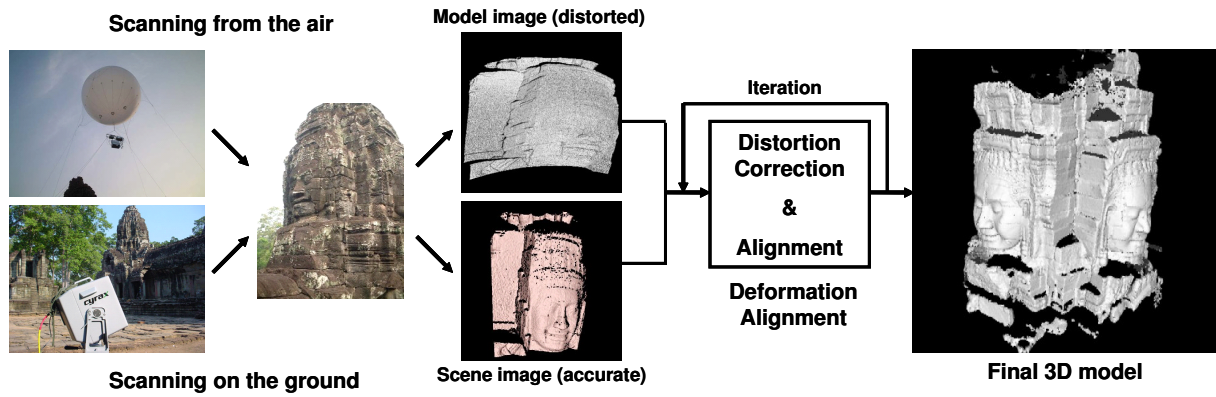


図 3.1: データ処理の流れ

表 3.1: CYRAX 2500 仕様

Device name	CYRAX 2500
Supplier	Cyra Inc. (USA)
Scanning length	1.5-50 m
Resolution of range	4 mm
Resolution of direction	60 μ radians
Sampling rate	maximum 1,000 pixel/sec

サの計測中の運動を記述するパラメータを同時推定する。



圖 3.2: CYRAX 2500 外觀

第4章

ハードウェア

本章では提案システムにおいて実際にデータを取得する気球搭載型センサについて関連研究と開発されたセンサの仕様を述べる。

4.1 関連研究

気球搭載型センサの詳細について述べる前に、本節では空中からの3次元データ取得に関する関連研究と、本研究で空中からモデリングを行うプラットフォームとして気球を採用した経緯について述べる。

空中からの計測の先行研究として、ヘリコプター、航空機を用いた地表形状計測、街のモデリングがある [3, 4, 5, 6]。しかし、これらの研究は計測対象物のはるか上空から物体の概観を計測することが目的であり、対象物の精緻な形状データを取得するために文化財に接近して計測するには危険度が高く、不向きである。仮にヘリコプターで寺院に接近できたとしてもエンジン、プロペラの細かい振動が計測結果に致命的な影響を及ぼす。対象物に安全に接近することができ、かつ致命的な振動による影響を受けない計測方法として、気球を用いる手法が考えられる。気球を用いた3次元形状の計測としてはデジタルカメラを用いたステレオ手法 [7] があるが、この手法では文化財のようにデータの精緻さが求められる場合には不十分である。

そこで本研究ではプラットフォームとしては気球を採用し、メソッドとしてはレーザーレンジセンサを採用した気球搭載型レーザーレンジセンサを開発することにした。

4.2 気球搭載型センサ

気球搭載型センサはスキャナーユニット、コントローラ、パーソナルコンピュータ(PC)から構成される(図4.1)。

スキャナーユニットはレーザーを実際に扱う部分で、気球の揺れの影響を最小限に抑えるためスキャンニングの高速化に最も重点を置いて設計した。図 4.2 にスキャナーユニット内部の様子を示す。スキャナーユニットは主にスポットレーザーレーダーユニットと二つの鏡から構成される。我々はこのレーザーレーダーユニットとして、サンプリングレートが他社と比較して高い(最大 625,000 pixel/sec)Z+F 社の LARA 25200 を採用した(表 4.1)。このレーザーレーダーユニットから発せられるレーザーの方向を二つの鏡が決定する。ユニット内部には図 4.2 にあるように 4 面のポリゴンミラーと平面鏡(ここではスイングミラーと呼ぶ)が設置され、前者がレーザーを水平方向角を決定し、後者が垂直方向角を決定する。1 フレームは 1 秒間で計測される。ポリゴンミラーは 2,400rpm で回転し、スイングミラーは 1 フレームで画像中心を 0 deg として -15 deg から +15 deg まで回転する。LARA 25200 のサンプリングレートは 576,000 pixel/sec に設定し、結果として 1 フレーム =(縦 160 pixel× 横 3,600 pixel) を 1 秒間で計測する(表 4.2)。

コントローラは信号処理ユニットとインターフェースユニットから構成される。信号処理ユニットは PC から送られてくる信号をスキャナーのレーザーレーダーユニットやミラーの角度等を決定する信号に変換してセンサの実際の制御を行う。また、レーザーレーダーユニットが得た距離とミラーの角度データはインターフェースユニットで順に結合され PC に距離画像として保存される。

以下に具体的な気球センサによる計測手順を示す。

1. 気球にヘリウムガスを注入する(図 4.3)。
2. 気球操作員(5-6 人)は気球をロープで操作する。気球にセンサを取り付けた後、徐々に気球を上昇させる(図 4.4)。
3. 対象物体を計測できる位置まで気球を移動させる(図 4.5)。
4. 地上にある PC で気球センサの PC をリモートコントロールし、コントローラにスキャンニング命令を送る。
5. 必要なデータの計測後、操作員は周囲の安全に気をつけながらロープを引きゆっくりと気球の高度を下げていく。
6. センサを取り外す。
7. 気球からヘリウムを抜き取る。このとき操作員はヘリウムガスを吸引しないように注意が必要である。

実際に気球センサによって得られたデータを図 4.6 に示す。図中にある青色の四角いグリッドの中心はセンサの持つ座標系の原点を意味する。

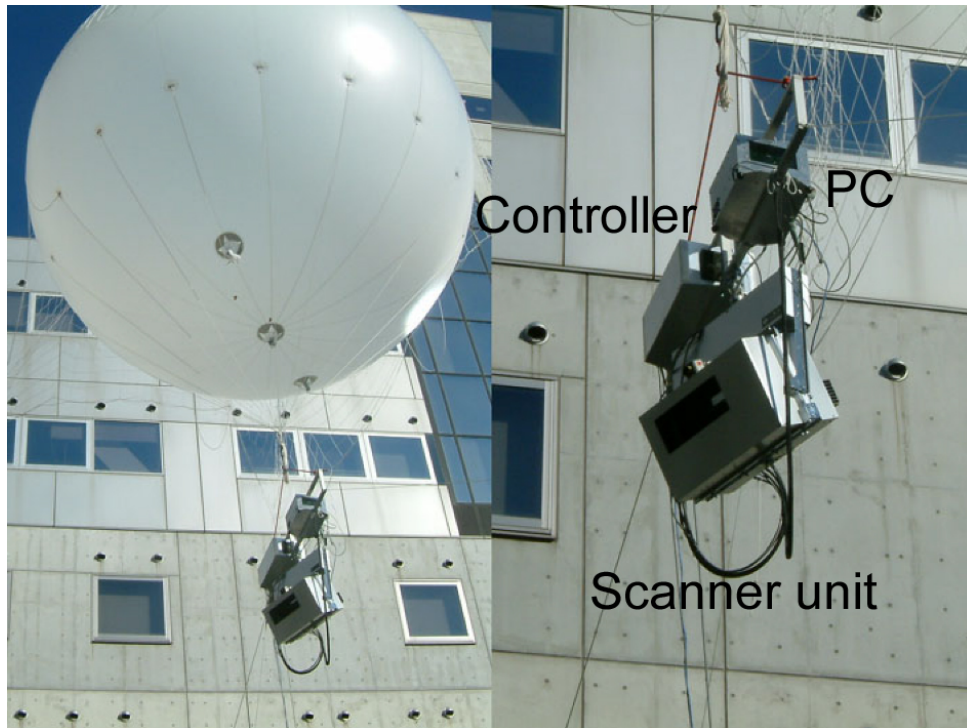


図 4.1: 気球センサ外観

表 4.1: LARA 25200 仕様

Device name	LARA 25200
Supplier	Z+F Inc. (Germany)
Light source	23 mW, 780 nm
Scanning length	2-25.2 m
Range resolution	0.38 mm
Linearity	less than 3 mm
Temperature drift	less than 2 mm
Beam diameter	3-4 mm
Sampling rate	maximum 625,000 pixel/sec

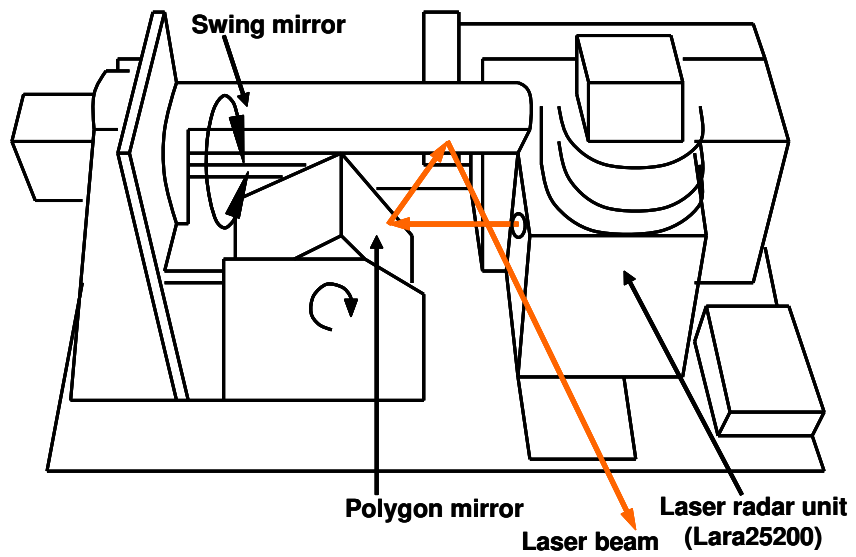
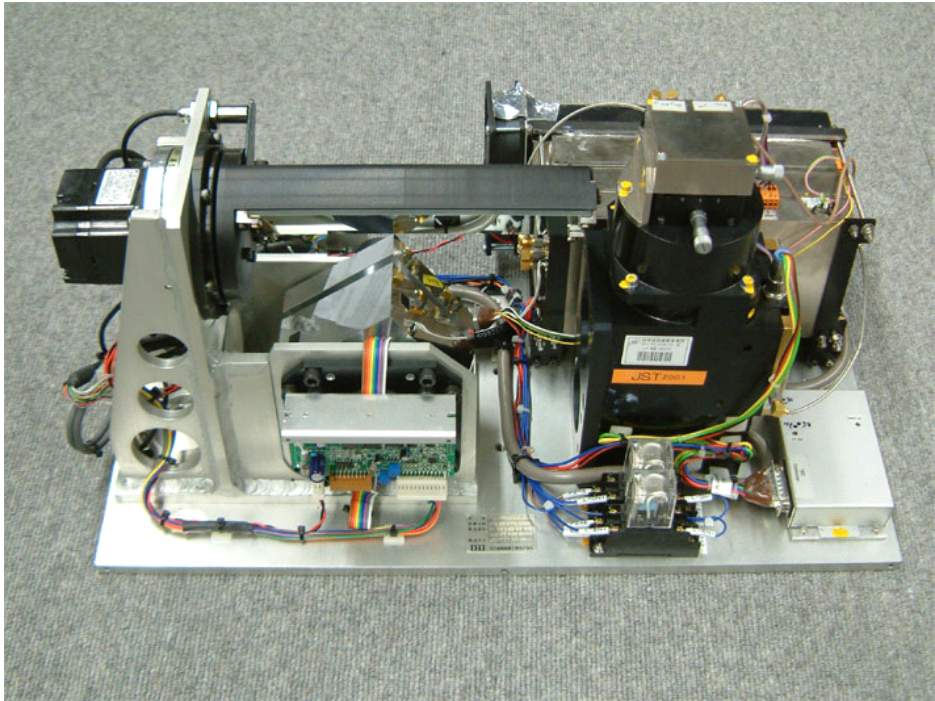


図 4.2: スキャナーユニット内部

表 4.2: 気球搭載型センサ (FLRF) 仕様

Vertical resolution	160 pixel
Horizontal resolution	3,600 pixel
Rotation speed (polygon mirror)	2,400 rpm
Swing range (swing mirror)	-15 deg - +15 deg
Scanning time for a frame	1 sec
Weight	21.2 kg
Size (depth × width × height)	27 cm × 53 cm × 30 cm



図 4.3: ヘリウムガスの注入



図 4.4: センサの取り付け



図 4.5: 目的の箇所を計測できる位置まで気球センサを操作

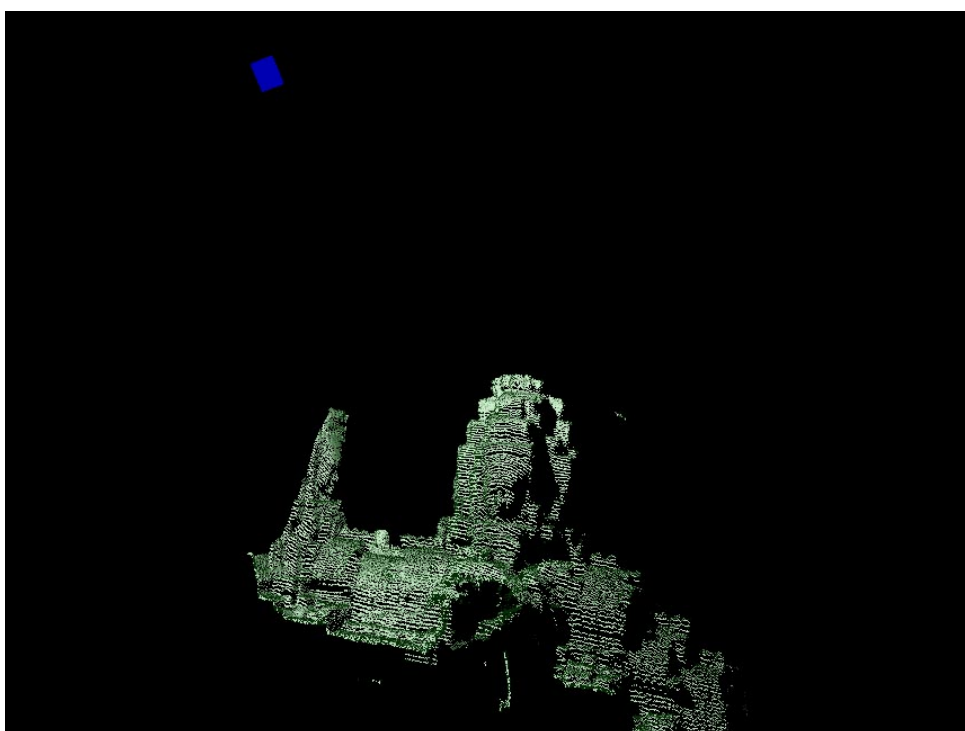


図 4.6: 気球センサによって得られた計測データ

第 5 章

ソフトウェア

気球では，ヘリコプターでエンジン等によって生じるような細かい振動はないものの，風等の外力による低周波で振幅の大きい揺れが生じ，計測された画像は歪んでしまう．本研究ではこの歪みをソフトウェアにより補正する手法を提案する．提案手法は地上の安定した地点から得られた別のデータに気球搭載型センサで得られた距離画像をレジストレーションする際にこの歪みを補正する．本章ではまず，このセンサの位置姿勢・運動パラメータ同時推定レジストレーションの理論について述べる．その後，本手法の推定精度を上げるためのアルゴリズムの改良について考察し，最後に本手法の推定精度をアルゴリズムの改良効果も交えて評価する．

5.1 関連研究

3次元レジストレーションの最も基本的なものとして Iterative Closest Point (ICP) アルゴリズムがある [8, 9]．これは対応点間の距離の総和を誤差関数とし，繰り返し計算により最小化ことで3次元データ間の位置姿勢を合わせるアルゴリズムである．ICP アルゴリズムを基本とした位置合わせ手法は現在，さまざまに拡張されているが，これらは主に (1) 位置合わせ順序，(2) 対応要素，(3) 誤差尺度の三つの観点から分類できる．

3次元データの位置合わせ順序には，逐次位置合わせと同時位置合わせが提案されている．逐次位置合わせは，対応する2枚の距離画像間の位置合わせを逐次繰り返して全体の位置合わせとする方法であるが，全体の位置合わせが終了した際に，位置合わせ誤差が局所的に蓄積される場合がある [10]．同時位置合わせは対応する全ての距離画像を考慮して一斉に全ての位置姿勢を決定するので，位置合わせ誤差は全ての距離画像間で分散され，逐次位置合わせよりも高精度な位置合わせが可能となる．

ICP アルゴリズムにおける誤差関数の対象として，距離画像の幾何学的な特徴点を用いる方法 [11, 12] と，3次元点を用いる方法 [8, 13] がある．特徴点を用いる位置合わせは，

全ての特徴点に一対一対応が得られることを前提とし、この対応を収束の各段階で変化させずに誤差関数を最小化するため、常に正しい対応が得られない場合は正確な位置合わせができない。3次元点を用いる場合は、収束の各段階において対応関係をより確からしいものに更新するので、特徴点を用いる手法よりもより正確に位置合わせが行われることが報告されている [14]。

各対応点間の誤差尺度として、ユークリッド距離を用いない方法としては2つの距離画像間の点と面の距離を用いるものがある [15, 16]。得られた計測データには、レーザーの反射率を示すリフレクタンスの値や、テクスチャ用の色情報等が付加されていることがあるが、上記の誤差尺度に加えてこれらのリフレクタンスの値や、テクスチャ用の色情報も考慮する位置合わせ手法が提案されている [17]。

本研究における最終的な目的は、位置姿勢のパラメータと同時に気球センサの計測中の運動パラメータを推定することである。推定するパラメータが増え、位置姿勢・形状が変化する自由度は高くなる。このような状況下では対応点の選択はより確実に行える最近傍点を用いる方法が望ましいと考える。そこで本研究では対応点として最近傍点を採用し、さらにその対応点間距離にローレンツ関数を乗じてロバスト性を高めた西野らの手法 [18] をベースにして、本研究で提案する位置合わせ手法を考えていくことにした。西野らは誤差関数を以下のように設定した。

$$E(\mathbf{p}) = \frac{1}{N(M-1)} \sum_i^N \sum_j^M \rho(z_{ij}(\mathbf{p})). \quad (5.1)$$

$$\text{但し, } \mathbf{p} = (\mathbf{t}, \mathbf{q}), \quad (5.2)$$

$$z_{ij}(\mathbf{p}) = \|\mathbf{R}(\mathbf{q}) \mathbf{x}_i + \mathbf{t} - \mathbf{y}_{ij}\|^2, \quad (5.3)$$

$$\rho(z_{ij}(\mathbf{p})) = \log \left(1 + \frac{1}{2} z_{ij}(\mathbf{p})^2 \right), \quad (5.4)$$

N : モデル画像上の点数,

M : シーン画像数.

パラメータ \mathbf{p} は並進移動を表すベクトル \mathbf{t} と回転を表すクォータニオン \mathbf{q} から成る。 \mathbf{x}_i はモデル画像上の i 番目の点を表し、 \mathbf{y}_{ij} は複数のシーン画像中の j 番目の画像上で \mathbf{x}_i に対応する点を表す。そして $E(\mathbf{p})$ は対応点間距離の二乗にローレンツ関数を用いた M 推定を施した値 $\rho(z_{ij}(\mathbf{p}))$ のモデル画像とシーン画像との全ての対応点の組み合わせにおける平均として定義される。

この誤差関数 $E(\mathbf{p})$ についてこれを最小とする次の式を満たすパラメータ \mathbf{p}_{opt} を計算する .

$$\mathbf{p}_{opt} = \arg \min_{\mathbf{p}} E(\mathbf{p}). \quad (5.5)$$

この非線形最小化問題は最急勾配法をベースに解かれる . 誤差関数の 7 次元空間での最急勾配は , 以下のように計算される .

$$\begin{aligned} \frac{\partial E}{\partial \mathbf{p}} &= \frac{1}{N(M-1)} \sum_i^N \sum_j^M \frac{\partial \rho(z_{ij})}{\partial z_{ij}} \cdot \frac{\partial z_{ij}}{\partial \mathbf{p}} \\ &= \frac{1}{N(M-1)} \sum_i^N \sum_j^M w(z_{ij}) z_{ij} \frac{\partial z_{ij}}{\partial \mathbf{p}}, \end{aligned} \quad (5.6)$$

$$\text{但し, } w(z_{ij}) = \frac{1}{z_{ij}} \cdot \frac{\partial \rho(z_{ij})}{\partial z_{ij}}.$$

$\frac{\partial z_{ij}}{\partial \mathbf{p}}$ は計算の簡略化のために単位クォータニオン \mathbf{q}_l (単位行列に相当) で評価する [19] . このとき , 回転クォータニオンの勾配は ,

$$\left. \frac{\partial \mathbf{R}(\mathbf{q}) \mathbf{x}_i}{\partial \mathbf{q}} \right|_{\mathbf{q}_l} = -2\mathbf{C}(\mathbf{x}_i) = 2\mathbf{C}(\mathbf{x}_i)^T \quad (5.7)$$

となる . ここで $\mathbf{C}(\mathbf{x}_i)$ は歪み対称行列 ,

$$\mathbf{C}(\mathbf{a}) = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix}. \quad (5.8)$$

$$\text{但し, } \mathbf{a} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix},$$

であり , 外積計算を以下のように表現できる .

$$\mathbf{a} \times \mathbf{b} = \mathbf{C}(\mathbf{a}) \mathbf{b}. \quad (5.9)$$

よって 7 変数の勾配は以下のように計算できる .

$$\begin{aligned}\frac{\partial z_{ij}(\mathbf{p})}{\partial \mathbf{p}} &= 2(\mathbf{R}(\mathbf{q})\mathbf{x}_i + \mathbf{t} - \mathbf{y}_{ij}) \frac{\partial (\mathbf{R}(\mathbf{q})\mathbf{x}_i + \mathbf{t} - \mathbf{y}_{ij})}{\partial \mathbf{p}} \Big|_{\mathbf{q}_t} \\ &= \begin{bmatrix} 2(\mathbf{x}_i + \mathbf{t} - \mathbf{y}_{ij}) \\ -4\mathbf{x}_i \times (\mathbf{t} - \mathbf{y}_{ij}) \end{bmatrix}.\end{aligned}\quad (5.10)$$

実際には最小化を効率的に行うために、Fletcher-Reeves 法、Polak-Riviere 法による共役勾配法 [20, 21, 22] を用いている。勾配方向に対する移動量は、黄金分割による囲い込み手法によって決定される。また k-d tree と呼ばれる、最近傍点探索を高速化するための、座標点の木構造の索引 [23, 24] を用いることにより対応点探索を高速に行う等の工夫もなされている。

5.2 位置姿勢と運動パラメータの同時推定手法

本研究で提案する位置姿勢と運動パラメータの同時推定手法について説明する。本手法は前節で紹介した位置姿勢パラメータの推定と同時に気球センサの歪みを生じさせる原因であるセンサの運動を記述するパラメータを同時に推定するものである。従ってまず、最初にするべきことは気球センサの運動をどのようなパラメータでどのように記述するかを規定することである。気球センサの計測中の運動は次の要素から成り立つ。

- 初期速度
- 初期角速度
- 外力による並進加速度
- 外力が発生させるモーメントによる角加速度

気球センサが 1 フレームを計測するのに要する時間が 1 秒と短いので、これらの要素のうちの加速度に関するものは無視できる。さらに角速度はそれが微小な場合、計測される距離画像に及ぼす影響は並進速度の影響として近似することができる。以上の理由からまず我々は気球センサの運動を一定の速度ベクトル \mathbf{v} による等速直線運動として次のような線形な式で記述した。

$$\mathbf{g}(\mathbf{v})_i = \mathbf{x}_i - \tau_i \mathbf{v}, \quad (5.11)$$

ここで τ_i はスキャンを開始してから点 \mathbf{x}_i が計測されたときに経過していた時間である。この値はミラーの回転速度等が既知であるために容易に導くことができる。この関数によって式 (5.21), (5.3) は気球センサの運動を考慮して次のように再定義される。

$$\mathbf{p} = (\mathbf{t}, \mathbf{q}, \mathbf{v}), \quad (5.12)$$

$$z_{ij}(\mathbf{p}) = \|\mathbf{R}(\mathbf{q})\mathbf{g}(\mathbf{v})_i + \mathbf{t} - \mathbf{y}_{ij}\|^2. \quad (5.13)$$

この定義をもとに式 (5.10) を再び導出すると次のようになる .

$$\begin{aligned} \frac{\partial z_{ij}(\mathbf{p})}{\partial \mathbf{p}} &= 2(\mathbf{R}(\mathbf{q})\mathbf{g}(\mathbf{v})_i + \mathbf{t} - \mathbf{y}_{ij}) \frac{\partial (\mathbf{R}(\mathbf{q})\mathbf{g}(\mathbf{v})_i + \mathbf{t} - \mathbf{y}_{ij})}{\partial \mathbf{p}} \Big|_{\mathbf{q}} \\ &= \begin{bmatrix} 2(\mathbf{g}(\mathbf{v})_i + \mathbf{t} - \mathbf{y}_{ij}) \\ -4\mathbf{g}(\mathbf{v})_i \times (\mathbf{t} - \mathbf{y}_{ij}) \\ 2(\mathbf{g}(\mathbf{v})_i + \mathbf{t} - \mathbf{y}_{ij}) \frac{\partial \mathbf{g}(\mathbf{v})_i}{\partial \mathbf{v}} \end{bmatrix}. \end{aligned} \quad (5.14)$$

ここで $\partial \mathbf{g}(\mathbf{v})_i / \partial \mathbf{v}$ は次のようになる .

$$\frac{\partial \mathbf{g}(\mathbf{v})_i}{\partial \mathbf{v}} = - \begin{bmatrix} (\tau_i \ 0 \ 0)^T \\ (0 \ \tau_i \ 0)^T \\ (0 \ 0 \ \tau_i)^T \end{bmatrix}. \quad (5.15)$$

5.3 実装

同時推定レジストレーションのアルゴリズムを Algorithm 1 に示す . 入力は気球センサから得られた 1 枚の距離画像 (モデル画像) d_c , 他のセンサで安定した地点から計測された複数の距離画像 (シーン画像) D , 気球センサの画像に与える初期パラメータ p_c , 他のセンサの画像それぞれに与える初期パラメータ P . 出力はモデル画像の位置合わせされたときのパラメータ p'_c , シーン画像の位置合わせされたときのパラメータ P' である . このほかにローカルにシーン画像それぞれに作成する k-d tree を格納するメモリ領域 $KDTree$ を用意する .

外側の repeat ループの処理ではまず始めにシーン画像の k-d tree を作成する . 次に for ループでシーン画像それぞれについて当該画像以外のシーン画像に対して誤差関数を最小とするパラメータを求めるだけで実際の反映はしない . さらにモデル画像をシーン画像に対して誤差関数を最小化するパラメータを求める . その後 , シーン画像とモデル画像にパラメータを反映させて移動・変形させる . これを誤差関数が収束するまで繰り返す .

次に *MinimizeErrorFunction* のアルゴリズムを Algorithm 2 に示す . *ComputeGradient* は式 (5.6) によって誤差関数のパラメータ p における勾配を求める . 求められた $gradient_i$ はさ

Algorithm 1 SimultaneousRegistrationConsideringParametersOfMovement

input: *DataByFlrf* d_c

input: *DataByOtherSensors* $D = \{d_i \mid i = 0, 1, \dots, n\}$

input: *InitialParametersOfPositionAndVelocityForFlrfData*

$$p_c = (R_c, t_c, v)$$

input: *InitialParametersOfPositionForOtherData* $P = \{p_i = (R_i, t_i) \mid i = 0, 1, \dots, n\}$

output: *AlignedPositionAndEstimatedVelocityForFlrfData*

$$p'_c = (R'_c, t'_c, v')$$

output: *AlignedPositionForOtherSensors*

$$P' = \{p'_i = (R'_i, t'_i) \mid i = 0, 1, \dots, n\}$$

local: *KDTree* = $\{kdt_i \mid i = 0, 1, \dots, n\}$

repeat

$KDTree \leftarrow MakeKDTree(P, D)$

for all $i = 0, 1, \dots, n$ **do**

$KDTree' \leftarrow KDTree - \{kdt_i\}$

$p'_i \leftarrow MinimizeErrorFunction(p_i, d_i, KDTree')$

end for

$p'_c \leftarrow MinimizeErrorFunction(p_c, d_c, KDTree)$

$P \leftarrow ChangePosition(P', P)$

$p_c \leftarrow ChangePositionAndParameter(p'_c, p_c)$

until *ErrorFunctionConverged*

$P' \leftarrow P$

$p'_c \leftarrow p_c$

Algorithm 2 MinimizeErrorFunction

input: *RangeData* d input: *InitialParameters* p input: *KDTree*output: *EstimatedParameters* p' local: *TempolaryMovedData* d' $gradient'_0 \leftarrow \mathbf{0}$ **for** $i = 1$ to *NumberOfIterations* **do** $d' \leftarrow \text{ChangePosition}(p)$ $gradient_i \leftarrow \text{ComputeGradient}(d', p)$ $gradient'_i \leftarrow \text{ConjugateGradient}(gradient_i, gradient_{i-1})$ $\lambda \leftarrow \text{LineMinimization}(d', gradient'_i)$ $p \leftarrow p + \lambda \cdot gradient'_i$ **end for** $p' \leftarrow p$

らに *ConjugateGradient* によりそれまでに求められた変位方向に共役になるように方向を修正され, その方向において *LineMinimization* で実際にパラメータを動かす変位量を決める. *LineMinimization* は誤差関数の変化させる方向での断面を二次関数であると仮定してこれを最も最小化する値を再帰的に求める. なお *ConjugateGradient* と *LineMinimization* についての詳細は [20, 21, 22] を参照すべし. 次に方向と変位 λ と $gradient'_i$ によりパラメータ p を更新する. アルゴリズムはこの一連の処理を適当な回数繰り返す. この回数は誤差関数が理想的な二次形式の場合パラメータ p の次元数繰り返すことで真の解に収束するが, 実際の誤差関数の形状はこの仮定を満たさない. そこで本実装では探索をパラメータの次元数である 10 回を反復した後に一度 $gradient'_{i-1}$ を 0 として 2 周目を探索している. 2 周目に 5 回探索して合計, 15 回としている.

5.4 精度の向上についての考察

5.4.1 評価式の改良

本節では同時推定レジストレーションの精度の向上について考察する. 後の 6.2.1 節の間引き影響評価でも述べるが, 本手法は従来の 7 パラメータのレジストレーションに比べて反復回数を要する. 共役勾配法の計算は誤差関数を最小化する唯一の解を探索する. し

かし，この解を求める反復過程で求める勾配(式(5.14))は実際には他のパラメータに常に干渉を及ぼしてしまい，最適な探索方向をとるとは限らない．本論文で提案している同時推定レジストレーションに限らず従来のレジストレーションにおいても，一般に最適化するパラメータ $\mathbf{R}(\mathbf{q})$ ， \mathbf{t} は互いに悪影響を及ぼしあっていることが知られている [18]．

本レジストレーションにおけるパラメータ同士が及ぼしあう悪影響を簡単な例を用いて説明する．いま図 5.1 左に示すような 4 点からなるモデル画像があるとする．5.2 節では，この画像は図 5.1 で示されるセンサの運動パラメータ \mathbf{v} が与えられると式(5.11)に従い，図 5.1 右上のように変化した．このとき各点で定義されているスキャン開始からの経過時間 τ が 0 である左下の点は位置変化がない．

さて，このようなモデル画像を図 5.2 左で示すようなシーン画像に位置合わせする場合を考える．ここでは簡単のためモデル画像が正しく位置合わせされた場合，シーン画像に完全に重なる事とし，さらにモデル画像とシーン画像の対応点は正しく与えられているとする．このとき， $\mathbf{R}(\mathbf{q})$ ， \mathbf{t} の勾配は誤差関数である対応点の距離の二乗和を最小化するために，図 5.2 右上に示す位置を目指す方向をとる．しかし，この方向はパラメータ \mathbf{v} にとっては必ずしも望ましい方向ではない． \mathbf{v} にとって最も望ましい $\mathbf{R}(\mathbf{q})$ ， \mathbf{t} の勾配方向は，モデル画像において $\tau = 0$ である左下の点がシーン画像における対応点と重なる図 5.2 右下に示す位置を目指す方向である．しかし実際はまず， $\mathbf{R}(\mathbf{q})$ ， \mathbf{t} が及ぼす影響から図 5.2 右上に示す位置とはまったく関係のない方向に移動してしまうし，仮に図 5.2 右上に示す位置に来たとしても次の反復計算で $\mathbf{R}(\mathbf{q})$ ， \mathbf{t} から影響を受け，その位置を離れてしまう．

そこで，この悪影響を最小に抑える方法を考える．上の例でもしモデル画像が図 5.2 右上にある位置から即座にシーン画像の形状に変化できるとしたら $\mathbf{R}(\mathbf{q})$ ， \mathbf{t} にとって望ましい探索方向は \mathbf{v} にとっても望ましい方向となる．そこで，モデル画像に座標変換 $\mathbf{R}_o(\mathbf{v})$ ， $\mathbf{t}_o(\mathbf{v})$ を施し，式(5.11)に替わって図 5.1 右下に示すような変形を実現させることを考える．このとき， \mathbf{v} を反映させる式は，

$$\mathbf{x}'_i = \mathbf{R}_o(\mathbf{x}_i - \tau_i \mathbf{v}) + \mathbf{t}_o, \quad (5.16)$$

となる．ここで ε_i を

$$\varepsilon_i = \|\mathbf{x}'_i - \mathbf{x}_i\|, \quad (5.17)$$

とおき， $\mathbf{R}_o, \mathbf{t}_o$ を

$$\begin{aligned} \mathbf{R}_o, \mathbf{t}_o &= \arg \min_{\mathbf{R}, \mathbf{t}} \sum_i^N \varepsilon_i^2 \\ &= \arg \min_{\mathbf{R}, \mathbf{t}} \sum_i^N \|\mathbf{R}(\mathbf{x}_i - \tau_i \mathbf{v}) + \mathbf{t} - \mathbf{x}_i\|^2, \end{aligned} \quad (5.18)$$

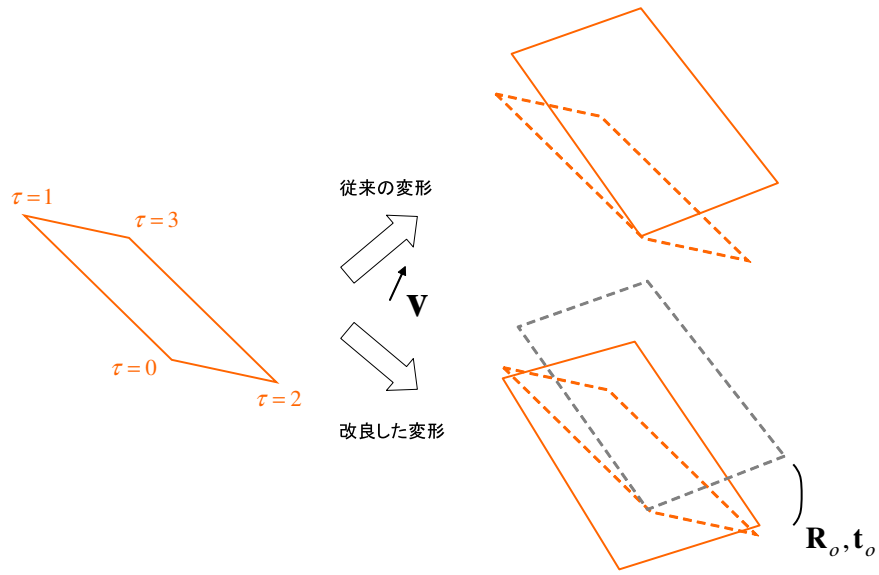


図 5.1: \mathbf{v} による形状変化

と定義すると，モデル画像は図 5.1 右下のように，画像の広がりをもその場に留めながら変形することを実現できる．以降， $\mathbf{R}_o(\mathbf{v}), \mathbf{t}_o(\mathbf{v})$ が具体的にどのように \mathbf{v} によって表されるのかを導出していく．

まず \mathbf{t}_o を，

$$\mathbf{t}_o = [t_{o1} \ t_{o2} \ t_{o3}]^T, \quad (5.19)$$

\mathbf{R}_o を線形近似するために歪み対称行列で

$$\mathbf{R}_o = \begin{bmatrix} 1 & -c_3 & c_2 \\ c_3 & 1 & -c_1 \\ -c_2 & c_3 & 1 \end{bmatrix}, \quad (5.20)$$

とし \mathbf{p} を次のようにおく．

$$\mathbf{p} = [t_{o1} \ t_{o2} \ t_{o3} \ c_1 \ c_2 \ c_3]^T. \quad (5.21)$$

そうすると式 (5.18) を満たす $\mathbf{R}_o, \mathbf{t}_o$ は，

$$\frac{\partial \sum_i^N \varepsilon_i^2}{\partial \mathbf{p}} = \sum_i^N 2\varepsilon_i \cdot \frac{\partial \varepsilon_i}{\partial \mathbf{p}} = \mathbf{0}, \quad (5.22)$$

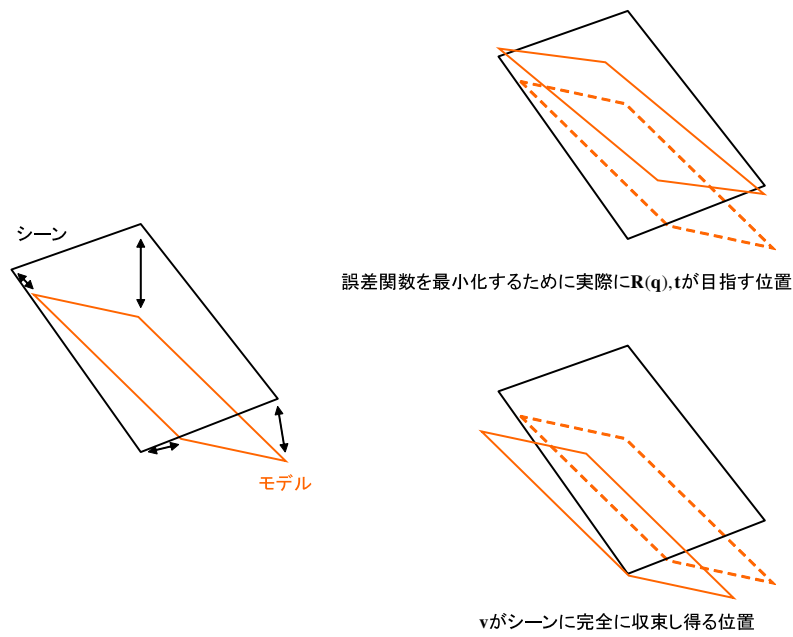


図 5.2: $\mathbf{R}(\mathbf{q}), \mathbf{t}$ が勾配にとる方向

の解となりこれを展開して解く. ε_i は,

$$\begin{aligned}
 \varepsilon_i &= \begin{bmatrix} 1 & -c_3 & c_2 \\ c_3 & 1 & -c_1 \\ -c_2 & c_3 & 1 \end{bmatrix} \begin{bmatrix} x_{i1} - v_1 \tau_i \\ x_{i2} - v_2 \tau_i \\ x_{i3} - v_3 \tau_i \end{bmatrix} + \begin{bmatrix} t_{o1} \\ t_{o2} \\ t_{o3} \end{bmatrix} - \begin{bmatrix} x_{i1} \\ x_{i2} \\ x_{i3} \end{bmatrix} \\
 &= \begin{bmatrix} -v_1 \tau_i - c_3(x_{i2} - v_2 \tau_i) + c_2(x_{i3} - v_3 \tau_i) + t_{o1} \\ c_3(x_{i1} - v_1 \tau_i) - v_2 \tau_i - c_1(x_{i3} - v_3 \tau_i) + t_{o2} \\ -c_2(x_{i1} - v_1 \tau_i) + c_1(x_{i2} - v_2 \tau_i) - v_3 \tau_i + t_{o3} \end{bmatrix}, \tag{5.23}
 \end{aligned}$$

より,

$$\begin{aligned}
 \varepsilon_i \cdot \frac{\partial \varepsilon_i}{\partial \mathbf{p}} &= \begin{bmatrix} \varepsilon_i \cdot \frac{\partial \varepsilon_i}{\partial t_{o1}} \\ \varepsilon_i \cdot \frac{\partial \varepsilon_i}{\partial t_{o2}} \\ \varepsilon_i \cdot \frac{\partial \varepsilon_i}{\partial t_{o3}} \\ \varepsilon_i \cdot \frac{\partial \varepsilon_i}{\partial c_1} \\ \varepsilon_i \cdot \frac{\partial \varepsilon_i}{\partial c_2} \\ \varepsilon_i \cdot \frac{\partial \varepsilon_i}{\partial c_3} \end{bmatrix} \\
 &= \begin{bmatrix} \begin{bmatrix} -v_1 \tau_i - c_3(x_{i2} - v_2 \tau_i) + c_2(x_{i3} - v_3 \tau_i) + t_{o1} \\ c_3(x_{i1} - v_1 \tau_i) - v_2 \tau_i - c_1(x_{i3} - v_3 \tau_i) + t_{o2} \\ -c_2(x_{i1} - v_1 \tau_i) + c_1(x_{i2} - v_2 \tau_i) - v_3 \tau_i + t_{o3} \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \\ \begin{bmatrix} -v_1 \tau_i - c_3(x_{i2} - v_2 \tau_i) + c_2(x_{i3} - v_3 \tau_i) + t_{o1} \\ c_3(x_{i1} - v_1 \tau_i) - v_2 \tau_i - c_1(x_{i3} - v_3 \tau_i) + t_{o2} \\ -c_2(x_{i1} - v_1 \tau_i) + c_1(x_{i2} - v_2 \tau_i) - v_3 \tau_i + t_{o3} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \\ \begin{bmatrix} -v_1 \tau_i - c_3(x_{i2} - v_2 \tau_i) + c_2(x_{i3} - v_3 \tau_i) + t_{o1} \\ c_3(x_{i1} - v_1 \tau_i) - v_2 \tau_i - c_1(x_{i3} - v_3 \tau_i) + t_{o2} \\ -c_2(x_{i1} - v_1 \tau_i) + c_1(x_{i2} - v_2 \tau_i) - v_3 \tau_i + t_{o3} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\ \begin{bmatrix} -v_1 \tau_i - c_3(x_{i2} - v_2 \tau_i) + c_2(x_{i3} - v_3 \tau_i) + t_{o1} \\ c_3(x_{i1} - v_1 \tau_i) - v_2 \tau_i - c_1(x_{i3} - v_3 \tau_i) + t_{o2} \\ -c_2(x_{i1} - v_1 \tau_i) + c_1(x_{i2} - v_2 \tau_i) - v_3 \tau_i + t_{o3} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ -(x_{i3} - v_3 \tau_i) \\ x_{i2} - v_2 \tau_i \end{bmatrix} \\ \begin{bmatrix} -v_1 \tau_i - c_3(x_{i2} - v_2 \tau_i) + c_2(x_{i3} - v_3 \tau_i) + t_{o1} \\ c_3(x_{i1} - v_1 \tau_i) - v_2 \tau_i - c_1(x_{i3} - v_3 \tau_i) + t_{o2} \\ -c_2(x_{i1} - v_1 \tau_i) + c_1(x_{i2} - v_2 \tau_i) - v_3 \tau_i + t_{o3} \end{bmatrix} \cdot \begin{bmatrix} x_{i3} - v_3 \tau_i \\ 0 \\ -(x_{i1} - v_1 \tau_i) \end{bmatrix} \\ \begin{bmatrix} -v_1 \tau_i - c_3(x_{i2} - v_2 \tau_i) + c_2(x_{i3} - v_3 \tau_i) + t_{o1} \\ c_3(x_{i1} - v_1 \tau_i) - v_2 \tau_i - c_1(x_{i3} - v_3 \tau_i) + t_{o2} \\ -c_2(x_{i1} - v_1 \tau_i) + c_1(x_{i2} - v_2 \tau_i) - v_3 \tau_i + t_{o3} \end{bmatrix} \cdot \begin{bmatrix} -(x_{i2} - v_2 \tau_i) \\ x_{i1} - v_1 \tau_i \\ 0 \end{bmatrix} \end{bmatrix}. \quad (5.24)
 \end{aligned}$$

以上の結果より式 (5.22) を解くと,

$$\mathbf{p} =$$

$$\begin{bmatrix} N & 0 & 0 & 0 & \sum_i^N X_3 & -\sum_i^N X_2 \\ 0 & N & 0 & -\sum_i^N X_3 & 0 & \sum_i^N X_1 \\ 0 & 0 & N & \sum_i^N X_2 & -\sum_i^N X_1 & 0 \\ 0 & -\sum_i^N X_3 & \sum_i^N X_2 & \sum_i^N (X_2^2 + X_3^2) & -\sum_i^N X_1 X_2 & -\sum_i^N X_1 X_3 \\ \sum_i^N X_3 & 0 & -\sum_i^N X_1 & -\sum_i^N X_1 X_2 & \sum_i^N (X_1^2 + X_3^2) & -\sum_i^N X_2 X_3 \\ -\sum_i^N X_2 & \sum_i^N X_1 & 0 & -\sum_i^N X_1 X_3 & -\sum_i^N X_2 X_3 & \sum_i^N (X_1^2 + X_2^2) \end{bmatrix}^{-1} \begin{bmatrix} v_1 \tau_i \\ v_2 \tau_i \\ v_3 \tau_i \\ v_3 \sum_i^N \tau_i X_2 - v_2 \sum_i^N \tau_i X_3 \\ v_1 \sum_i^N \tau_i X_3 - v_3 \sum_i^N \tau_i X_1 \\ v_2 \sum_i^N \tau_i X_1 - v_1 \sum_i^N \tau_i X_2 \end{bmatrix} \quad (5.25)$$

但し, $X_k = x_{ik} - v_k \tau_i$,

となり $\mathbf{R}_o, \mathbf{t}_o$ を \mathbf{v} で表すことができた．実際にここで得られた c_1, c_2, c_3 をそのまま式 (5.20) に代入して \mathbf{R}_o に戻したのでは徐々に画像が変形してしまう．そこで次の式を用いて c_1, c_2, c_3 から回転行列に正規化した \mathbf{R}_o を求める．

$$\mathbf{R}_o = \begin{bmatrix} \cos c_3 & \sin c_3 & 0 \\ \sin c_3 & \cos c_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos c_2 & 0 & \sin c_2 \\ 0 & 1 & 0 \\ \sin c_2 & 0 & \cos c_2 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos c_1 & \sin c_1 \\ 0 & \sin c_1 & \cos c_1 \end{bmatrix}. \quad (5.26)$$

しかし次に問題が発生する．導かれた式 (5.25) を本レジストレーションに実装する場合, 式 (5.14) において $\partial \mathbf{g}(\mathbf{v})_i / \partial \mathbf{v}$ を求める必要があるがこれは,

$$\frac{\partial \mathbf{g}(\mathbf{v})_i}{\partial \mathbf{v}} = \frac{\partial}{\partial \mathbf{v}} (\mathbf{R}_o (\mathbf{x}_i - \tau_i \mathbf{v}) + \mathbf{t}_o), \quad (5.27)$$

となり先ほどの式 (5.25) を \mathbf{v} で微分する必要が生じる．しかし式 (5.25) は異なる 3 点以上の数値を代入したときでないと \mathbf{R}_o を \mathbf{t}_o 一意的に定めることはできないため, 実際の値を持たないまま導関数を求めることはできないという壁に行き当たる．

ここでとりうる選択肢は 3 つ考えられる．

1. 式 (5.25) をコンピュータにより数値的に微分する．
2. 式 (5.16) の $\mathbf{R}_o, \mathbf{t}_o$ を既存の位置合わせアルゴリズムを用いてを求める．
3. 式 (5.14) において代替となる簡単な式を用いて近似的な勾配を求め, その値を用いる．

1. において問題となるのは勾配の値を実際に数値を代入して求める際の代入値のスケールであろう．本研究では微分する変数 \mathbf{v} は気球の速度ベクトルであり, 常におおよそ同じオーダーの値をとるのでスケールアップの問題は簡単に解決できるのでこの手法は有効であると考えられる．2. はこの場合対応点が完全に与えられているので誤差関数は理想的

な二次形式となり 5.1 節で紹介した既存の位置合わせ手法でパラメータ $(\mathbf{R}_o, \mathbf{t}_o)$ の次元数である 7 回の反復計算で高速に求めることができる。3. は式の関係が理論的には正しくなく、常に誤差を持つものであるが 1., 2. でアルゴリズムが複雑になってしまうことを考えればメリットは十分にあると考える。

本節ではこれらの選択肢のうち 3. に沿ってアルゴリズムの改良を試みる。

5.4.2 簡略化

本節では前節での改良手法を簡略化する。式 (5.25) が複雑になる原因は式 (5.16) において \mathbf{v} を含む項の積 $-\tau_i \mathbf{R}_o \mathbf{v}$ があり、これが簡単な形に変形できないことが一因にあげられる。ここでこの項を線形にするために \mathbf{R}_o を無視しても \mathbf{t}_o によってある程度の最適化がされる。そこでここでは式 (5.16) において $\mathbf{R}_o = \mathbf{I}$ とおいた次式、

$$\mathbf{x}'_i = \mathbf{x}_i - \tau_i \mathbf{v} + \mathbf{t}_o, \quad (5.28)$$

で $\partial \mathbf{g}(\mathbf{v})_i / \partial \mathbf{v}$ を求めることにする。このとき

$$\begin{aligned} \boldsymbol{\varepsilon}_i &= \mathbf{x}'_i - \mathbf{x}_i \\ &= -\tau_i \mathbf{v} + \mathbf{t}_o. \end{aligned} \quad (5.29)$$

さらに式 (5.22) を解くと

$$\mathbf{t}_o = \frac{\sum_i^N \tau_i}{N} \mathbf{v} \quad (5.30)$$

となる。この結果を用いて式 (5.28), (5.15) を書き直すと

$$\mathbf{x}'_i = \mathbf{x}_i - \left(\tau_i - \frac{\sum_i^N \tau_i}{N} \right) \mathbf{v}, \quad (5.31)$$

$$\frac{\partial \mathbf{g}(\mathbf{v})_i}{\partial \mathbf{v}} = - \begin{bmatrix} \left(\left(\tau_i - \frac{\sum_i^N \tau_i}{N} \right) 0 0 \right)^T \\ \left(0 \left(\tau_i - \frac{\sum_i^N \tau_i}{N} \right) 0 \right)^T \\ \left(0 0 \left(\tau_i - \frac{\sum_i^N \tau_i}{N} \right) \right)^T \end{bmatrix}, \quad (5.32)$$

と回転を無視した $\partial \mathbf{g}(\mathbf{v})_i / \partial \mathbf{v}$ が求まる。本節で述べたアルゴリズムは後の 6.2.3 節で比較検証していくことにする。

第 6 章

ソフトウェアの評価

本章では前節で述べたレジストレーション手法を実際に気球センサから得られたデータに適用した結果を示すと共に，人工的に合成された理想データを用いて様々な条件下で実験を行い本手法の精度と限界を定量的に評価する．

6.1 実データにおける実験

図 6.1 に示す黄色と紫色のデータはカンボジアのバイヨン寺院の物で黄色の点群が気球センサにより計測されたデータを示し，紫色の点群が地上に設定された CYRAX 2500 から得られたものである．紫色のデータをシーン画像，黄色のデータをモデル画像として前章で述べた同時推定レジストレーションをかけた結果を図 6.2 に示す．二つの図を見比べて分かるように黄色いデータは紫のデータに重なるように位置姿勢が動いているのと同時に，その形状もぴったり重なる形状に変化している．

しかしながらこの実験では残念ながら二つのデータがどの程度よく位置合わせされたのかは評価することができない．次節ではあらかじめ既知のパラメータで位置姿勢，形状を変化させたデータを用いてこれらのパラメータをどの程度よく推定できたのかを定量的に評価していく．

6.2 精度の評価

本節ではレジストレーション精度を定量的に評価するために CYRAX 2500 で得たデータ(図 6.3)を既知のパラメータで移動変形させてできた距離画像を用いて実験を行う．基本的な評価手順は以下のとおりである．

1. 距離画像を 1 枚とそのコピーを用意する(図 6.3)．

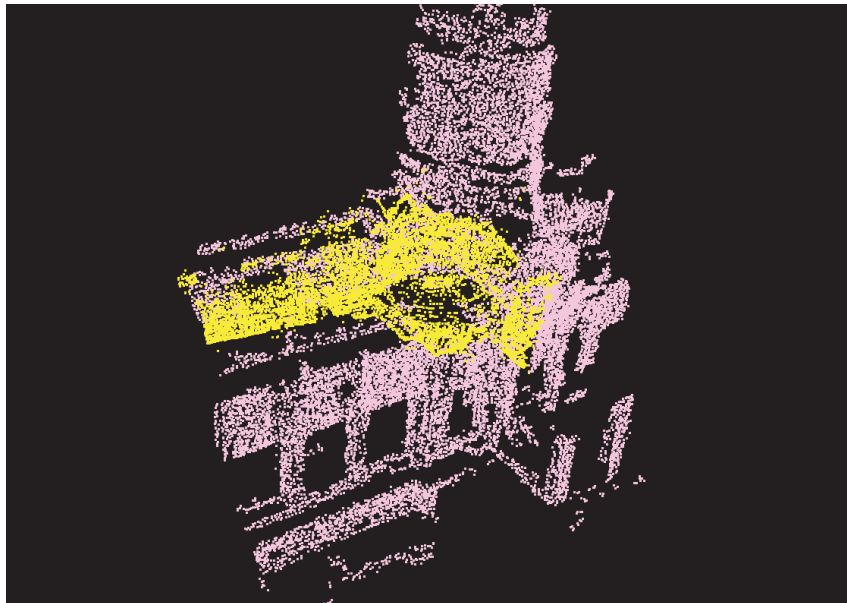


図 6.1: 実データ (レジストレーション前)

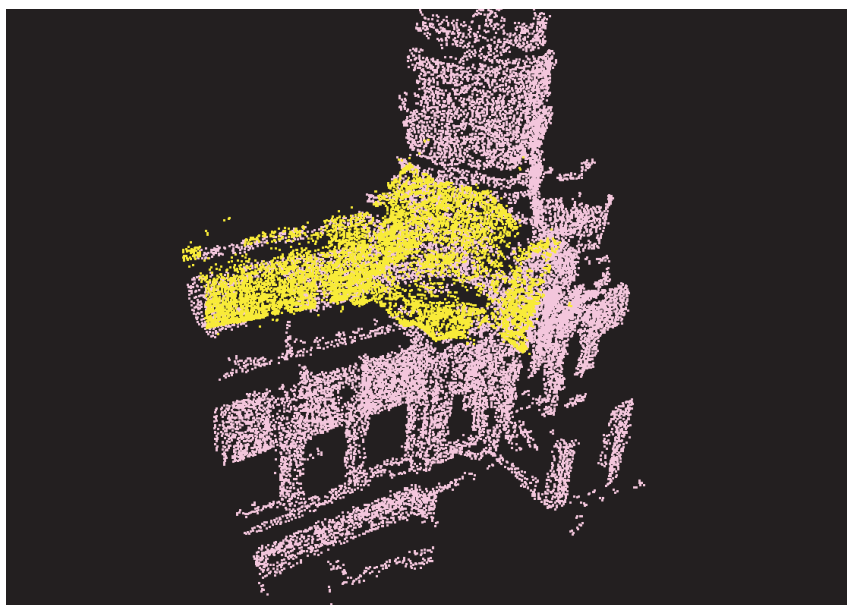


図 6.2: 実データ (レジストレーション後)

2. 片方の画像の左端から 20%の点を取り除きもう一方の画像は右端から同様に 20%の点を取り除く。(結果として 2 枚のデータの重なる領域は中央の 60%の領域となる.)
3. 2 枚の画像の点をランダムにそれぞれ間引く.
4. レジストレーションによって推定するパラメータ $(\mathbf{R}(\mathbf{q}), \mathbf{t}, \mathbf{v})$ の真値を設定しこのパラメータによって片方の画像を移動・変形する.
5. 移動・変形させた画像をモデル画像, もう片方をシーン画像として同時推定レジストレーションをかけ結果を記録する.
6. 設定する真値を必要に応じて変える.
7. 手順 4.-6. を必要回数, 繰り返す.

本節では本研究で提案した同時推定レジストレーション手法の計算コスト, 精度について評価していくがそれに先立ってレジストレーションに用いる点数の多少がレジストレーション結果に与える影響を調べる. というのも一般的にレーザーレンジセンサが 1 回の計測で得ることのできる座標点数は少なくとも数十万点と膨大な量である. これら無数の点について最近傍点探索を本レジストレーションの反復計算で何回も実行していくと計算時間が非常にかかる. (k -d tree を用いた 1 点についての最近傍点探索のオーダは $O(\log n)$) そこでレジストレーションの現実的な使用を考えると計算に用いる点をあらかじめ間引く必要がある. 本節ではこの点の間引きの影響評価, 提案手法の計算コスト, 提案手法のパラメータ推定精度の 3 点について評価を行う.

6.2.1 点の間引きの影響評価

図 6.3 で示したデータは 640,000 点からなるがこの点数の間引きによる計算時間・反復回数へ与える影響とパラメータ推定結果に与える影響を評価する. 実験において設定したパラメータの真値と計算にレジストレーションに用いる点数を表 6.1 にまとめた.

図 6.4, 6.5, 6.6 はそれぞれレジストレーションに用いた点数に対する計算に用いた点数に対する総計算時間, 計算における反復 1 回あたりに要した時間, 収束するまでに要した総反復回数を示す. 図では気球の運動 (物体の形状変化) を考慮しない従来手法と比較してある. 図 6.5 から分かるように反復計算 1 回あたりに要する時間はほとんど変わらない. それにもかかわらず図 6.4 の総計算時間でおよそ 5 倍ほど時間を要しているのはパラメータの次元数が高く, 自由度が高いために収束までに反復回数を要するからであると考えられる (図 6.6).



図 6.3: 評価に用いるデータ

図 6.7 は計算に用いた点数とレジストレーションによって得られたパラメータの値と真値との誤差を表している．誤差はそれぞれ translation(\mathbf{t}) と rotation($\mathbf{R}(\mathbf{q})$) , velocity(\mathbf{v}) に分けそれぞれの推定ベクトルと真値ベクトルとの差の大きさをグラフにしている．点数が極端に少ない領域では正しい解に収束していないが計算に用いる点が 7,000 点を超えたあたりから解の誤差は小さくなりそれ以降は安定してくる．この結果より以降の評価実験では計算する点として 8,000 点を用いて行うことにする．

表 6.1: 間引き影響評価実験条件

パラメータ	設定値
$\mathbf{R}(\mathbf{q})$	5 deg along X axis
\mathbf{t}	0.3 m along X axis
\mathbf{v}	0.3 m/s along X axis
計算点数	640,000 点中160-20,480 点

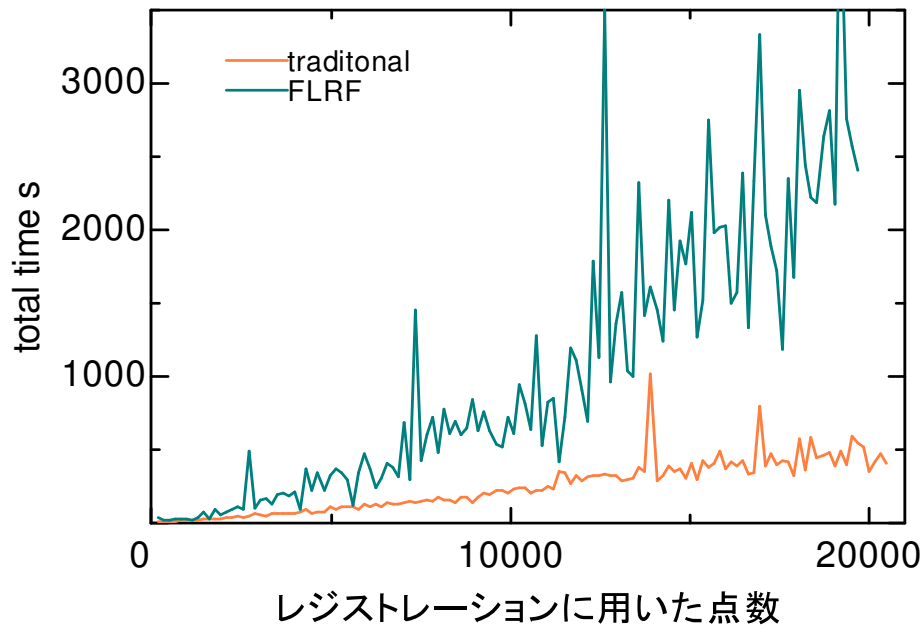


図 6.4: 間引き影響評価-収束するまでに要した総時間

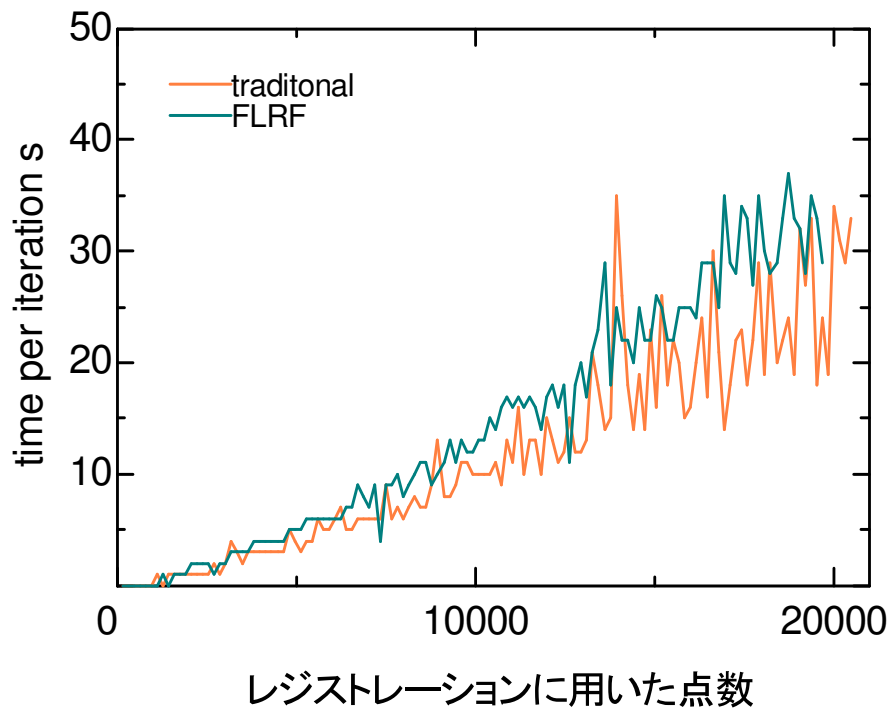


図 6.5: 間引き影響評価-反復 1 回当たりに要した時間

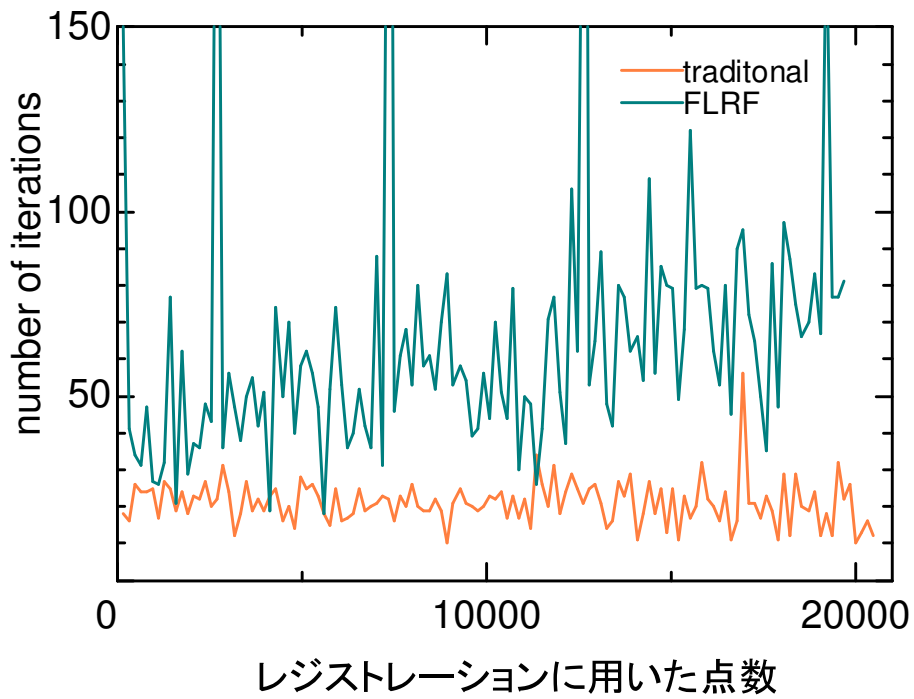


図 6.6: 間引き影響評価-収束するまでに要した反復回数

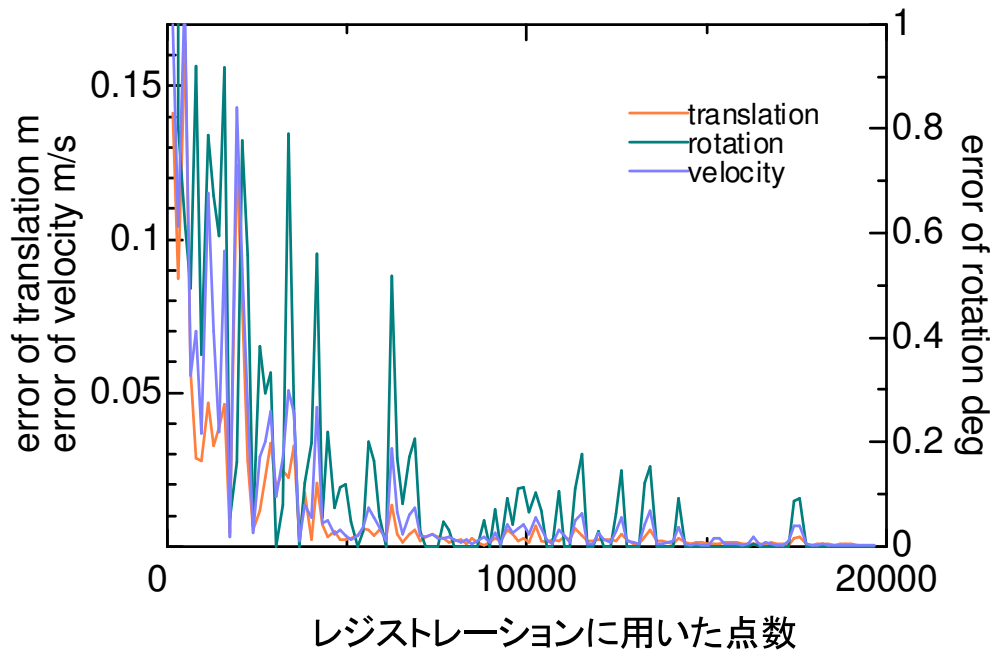


図 6.7: 間引き影響評価-推定パラメータの真値との誤差

表 6.2: 同時推定レジストレーションの精度評価実験条件

パラメータ	設定値
$\mathbf{R}(\mathbf{q})$	3 deg along X axis
\mathbf{t}	0.1 m along X axis
\mathbf{v}	0.00-3.00 m/s along X axis (0.01 m/s 刻み)

表 6.3: 同時推定レジストレーションのパラメータ推定精度

パラメータ	推定精度
\mathbf{t}	0.005 m 以下
$\mathbf{R}(\mathbf{q})$	0.1 deg 以下
\mathbf{v}	0.008 m/s 以下

6.2.2 同時推定レジストレーションの精度評価

ここでは5.2節で紹介した同時推定レジストレーションのパラメータ推定精度について評価する．実験条件を6.2にまとめた．実際にレジストレーションを行うときは位置姿勢についてはGUIを用いてある程度よい精度で初期値を与えることが期待される．このため本実験では先の実験よりも位置姿勢パラメータである $\mathbf{R}(\mathbf{q})$ ， \mathbf{t} はGUIでの手動設定誤差を想定した比較的小さい値(x軸に沿ってそれぞれ3 deg，0.1m)に固定した上で \mathbf{v} を変動させてこれに対するパラメータ推定誤差を見る．実験結果を図6.8，6.9，6.10に示す．実験は点の間引きによる偶然性を極力排除するためにそれぞれの設定値 \mathbf{v} でレジストレーションを5回行いそれぞれの誤差パラメータの最低値と最高値を切り捨て中央の3データについて平均をとったもので結果を作成した．

まず，本手法がおおよそ正しいパラメータ値を推定できる限界について見る．図6.8，6.9，6.10に共通して言えることは設定 velocity が1.6 m/s 付近まではどれも大きな収束誤差はない．しかしこの付近を超えると誤差は急激に増大し正しい解に収束することができなくなっている．本手法による気球の速度の限界は1.6 m/s 程度であると結論付けられる．

次におおよそ正しいパラメータを推定しているセンサ速度2.1 m/s 以下の領域におけるパラメータ推定精度について見る．図6.8より \mathbf{t} に関しては平均的な誤差は0.005 m 以下である．図6.9より $\mathbf{R}(\mathbf{q})$ に関しては平均的な誤差は0.1 deg 以下である．図6.10より \mathbf{v} に関しては平均的な誤差は0.008 m/s 以下である．以上の結果を表6.3にまとめる．

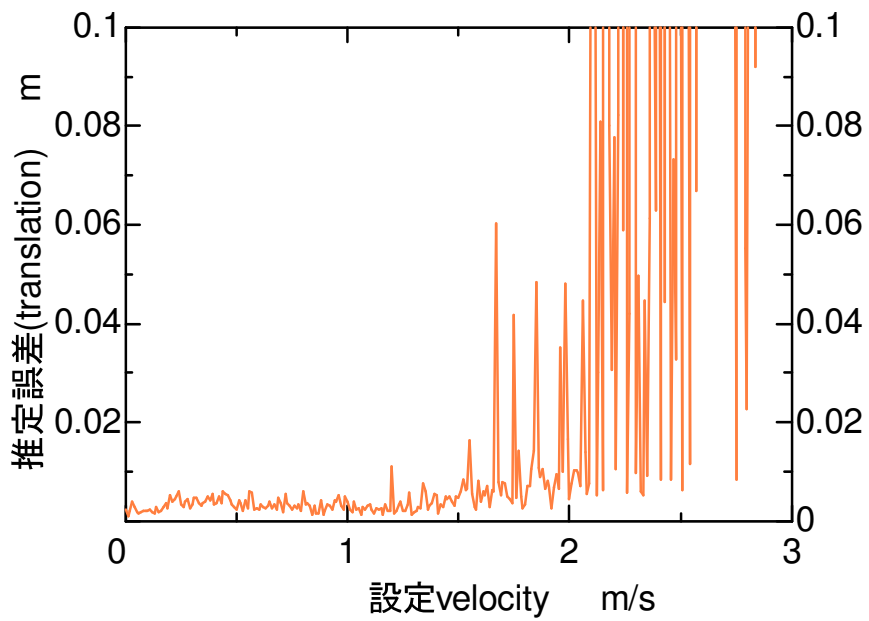


図 6.8: レジストレーション精度 (t)

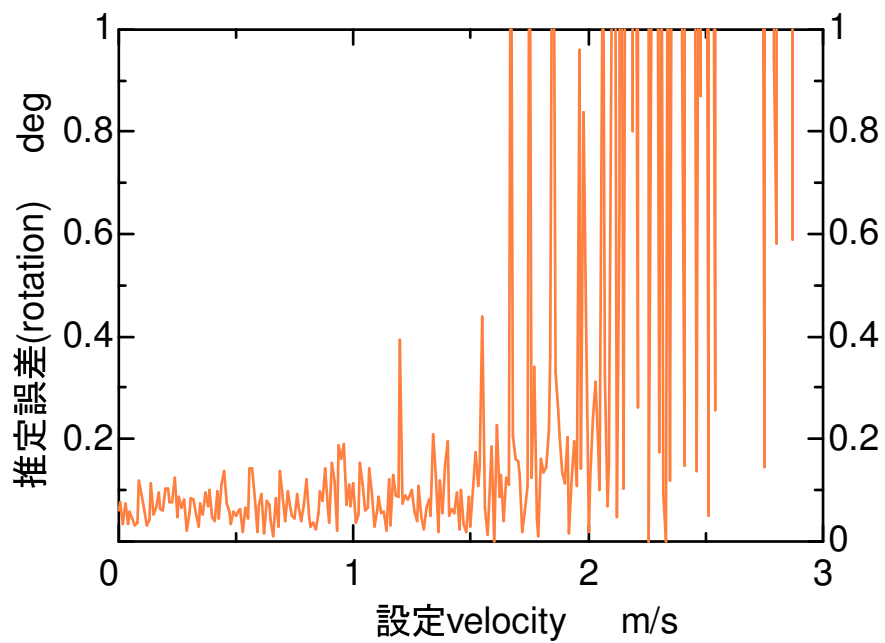


図 6.9: レジストレーション精度 ($\mathbf{R}(\mathbf{q})$)

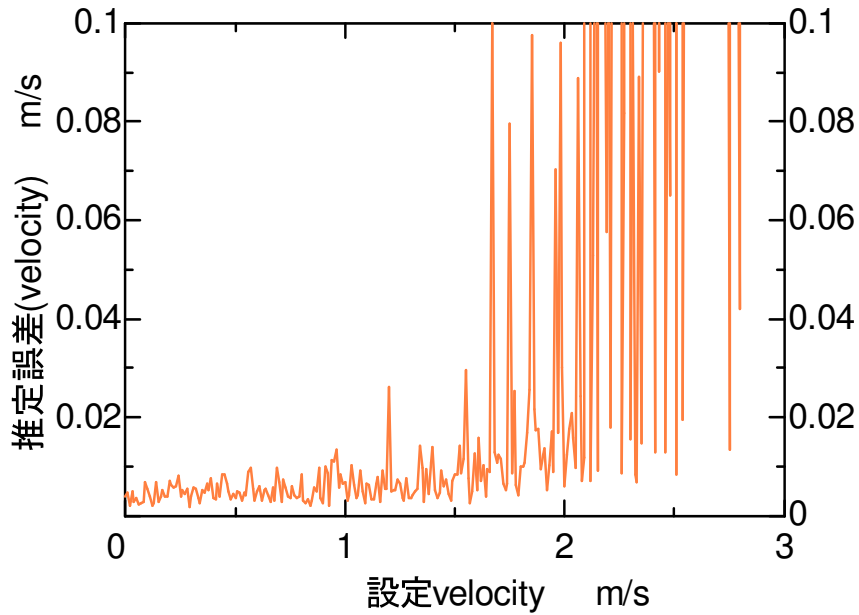


図 6.10: レジストレーション精度 (v)

6.2.3 改良アルゴリズムの比較実験

本節では改良アルゴリズムの推定精度を元の手法と比較するための実験を行う．実験は表 6.5 に示した 2 条件について行う．実験条件 A は v の反映式，その導関数，共に回転を無視して簡略化したものを用いる実験である．実験条件 B では v の反映式には回転 \mathbf{R}_0 を考慮した式 (5.16) を用いる．導関数には回転を無視して簡略化したものを用いる．実験条件 B は導関数は厳密には反映式の導関数ではないが，パラメータに従って実際に頂点を動かすときに $\mathbf{R}(\mathbf{q})$ への干渉を低くすることができるだろう．結果としては 1 回の反復計算での移動量が少ないものの無駄な動きが減り，安全性の高い (誤収束のない) レジストレーションを可能とすることが期待される．

実験条件 B では評価式として式 (5.16) を用いるためにレジストレーション中の至るところで必要とされる誤差関数の評価において毎回，式 (5.25) の総和計算を行わなくてはならずこれが計算コストとなる．しかし実はここで行う計算はあくまで反復法によりパラメータを求める前にパラメータ同士の干渉を軽減するための前処理であり，必ずしも厳密に行う必要はない．そこで本実験では式 (5.25) を計算するのに用いる適当に抽出された点群を決めておきこれを用いることで計算の高速化を図れる．そこでまず，予備実験として抽出する点数と実際の解にどのような関係があるのかを調べ，評価実験で用いる点数を決定する．実験は評価実験で計算に用いる総点数 8,000 点の中から均一に何点かを抽出した物を用いて，式 (5.25) を計算した結果を計算点を抽出せずに全ての点を計算に用いた結果と比較して，評価実験を行う際に用いる計算点数を決定する．実験結果を表 6.5 に示す．実験

表 6.4: 計算点数の決定のための予備実験結果

計算に用いた点数	t_1	t_2	t_3	c_1	c_2	c_3
8000/8000 点	0.187879	0.274053	0.253448	-0.022297	-0.076631	0.132765
1000/8000 点	0.185235	0.273645	0.250744	-0.024234	-0.070933	0.132996
500/8000 点	0.185510	0.273379	0.251810	-0.022515	-0.073206	0.133248
300/8000 点	0.181986	0.276300	0.250983	-0.028778	-0.079227	0.136251
100/8000 点	0.154873	0.272635	0.260308	-0.014865	-0.127596	0.142900

表 6.5: 改良アルゴリズム評価実験条件

実験条件	$g(\mathbf{v})_i$	$\partial g(\mathbf{v})_i / \partial \mathbf{v}$
A	式 (5.31)	式 (5.32)
B	式 (5.16)	式 (5.32)

結果より，比較的精度が高いにもかかわらず計算点数が少なくてすむ 500 点を評価実験における計算点数として採用することにする．(繰り返しになるがこの実験で算出される精度は評価実験において本質的な問題ではないのでここではこれ以上の議論は省略する．)

図 6.11 に改良アルゴリズム評価実験結果を示す．まず実験条件 B は残念ながら改良前のアルゴリズムよりも結果が悪くなっている．この原因としては実験条件 B における評価式 (5.16) の導関数として，実際の導関数とは異なる式 (5.32) を用いていることが原因と考えられる．一方で，実験条件 A は \mathbf{R}_0 を無視している点で最適化が完全に行われてはいないが理論的に式の対応はとれている (勾配の計算に実際の導関数を用いている) 点でよい効果が得られていると考えられる．実験条件 A では改良前のアルゴリズムが歪みを補正できる限界速度が 1.6 m/s 程度であったのを 2.6 m/s 程度まで引き上げることに成功している．

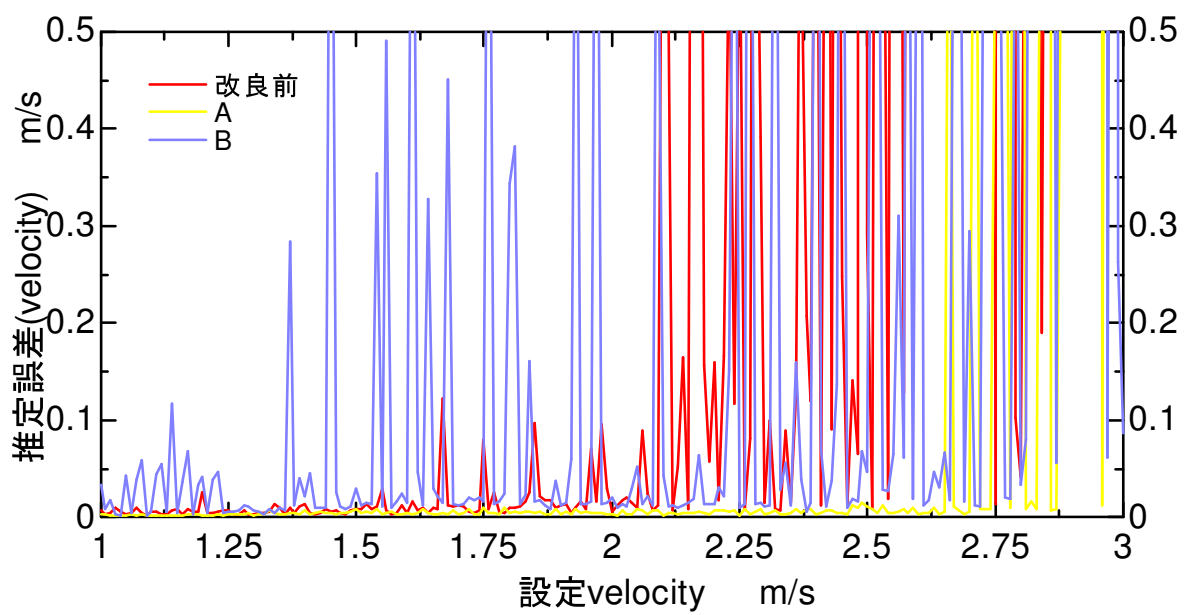


図 6.11: 改良アルゴリズム評価実験

第7章

まとめ

気球上から文化財の3次元データを取得し、そのデータの歪みを補正してモデリングを行うシステムを提案した。これを実現するために気球搭載型レーザーレンジセンサを開発し、また、センサから得られた画像の歪みを補正する、位置姿勢及び運動パラメータの同時推定レジストレーション手法を開発した。

これにより、文化財のモデリングにおいて地上からの計測ではオクルージョンとなってしまう領域を空中からの計測により補うことが可能となった。具体的には、気球搭載型レーザーレンジセンサの開発はステレオ手法等に比べて精度が高いレーザーレンジセンサによる距離画像を空中からも得ることを可能とした。また気球センサ自体が動いてしまうことから生じる距離画像の歪みを高い(センサの)許容速度と推定精度で補正することを可能とした。

将来の課題として、気球センサは使い勝手の向上があげられる。重量の軽量化、セッティングの簡便化、制御プログラムの改良をしていく予定である。レジストレーションとしては、5.4節でアルゴリズムの改良の指針を示し、一部については実験を行いその有効性を確認した。今後は、5.4.1節で実装方法として述べたものの、今回、評価していない数値微分による勾配計算、あるいは式(5.16)の \mathbf{R}_o, t_o を従来の位置合わせ手法で解いた場合の精度評価も行っていく。また本研究では気球センサの等速直線運動を仮定してレジストレーションを行ったが、このレジストレーションで収束させた後に、より厳密な近似を利用したレジストレーションを行うことで推定精度を向上させることも考えていく予定である。

参考文献

- [1] D. Miyazaki, T. Oishi, T. Nishikawa, R. Sagawa, K. Nishino, T. Tomomatsu, Y. Yakase, and K. Ikeuchi, “The great buddha project: Modelling cultural heritage through observation,” in *Proc. of the 6th Int. Conf. on Virtual Systems and MultiMedia*, October 2000, pp. 138–145.
- [2] M. Levoy et al, “The digital michelangelo project: 3d scanning of large statues,” in *ACM SIGGRAPH Proceedings*, July 2001, pp. 131–144.
- [3] Mark Daniel Sebastian Thrun and Dirk H ahnel, “Scan alignment and 3-d surface modelling with a helicopter platform,” in *The 4th Int. Conf. on Field and Service Robotics, July 14-16, 2003*.
- [4] Ryan Miller and Omead Amidi, “3-d site mapping with the cmu autonomous helicopter,” June 1998.
- [5] Tianen Chen and Ryosuke Shibasaki, “Ground truth measurement system using rc helicopter,” in *Proc. Asian Conf. on Remote Sensing*. Chiba University, 1999.
- [6] Xinhua Wang Armin Gruenl, Zhang Lil, “3d city modelling with tls (three-line scanner) data,” in *Int. Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XXXIV-5/W10.
- [7] Jana Visnovcova, Li Zhang, and Armin Gruen, “Generating a 3d model of a bayon tower using non-metric imagery,” in *Proc. of Int. Workshop Recreating the Past -Visualization and Animation of Cultural Heritage*, 2001.
- [8] P.J. Besl and N.D. McKay, “A method for registration of 3-d shapes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, February 1992.
- [9] Z.Y. Zhang, “Iterative point matching for registration of free form curves and surfaces,” *International Journal of Computer Vision*, vol. 13, no. 2, pp. 119–152, October 1994.

- [10] G. Turk and M. Levoy, “Zipped polygon meshes from range images,” in *ACM SIGGRAPH Proceedings*, July 1994, pp. 311–318.
- [11] K. Higuchi, M. Herbert, and K. Ikeuchi, “Building 3-d models from unregistered range images,” in *Graphical Models and Image Processing*, July 1995, vol. 57, pp. 315–333.
- [12] A.E. Johnson and M. Herbert, “Surface matching for object recognition in complex 3-dimensional scenes,” *Image and Vision Computing*, vol. 16, no. 9/10, pp. 635–651, July 1998.
- [13] David Simon, *Fast and Accurate Shape-Based Registration*, Ph.D. thesis, School of Computer Science, Carnegie Mellon University, 1996.
- [14] Szymon Rusinkiewicz and Marc Levoy, “Efficient variants of the icp algorithm,” in *Proc. of the 3rd Int. Conf. on 3D Digital Imaging and Modeling*, May 2001, pp. 145–152.
- [15] Y. Chen and G. Medioni, “Object modelling by registration of multiple range image,” in *Image and Vision Computing*, Apr 1992, vol. 10(3), pp. 145–155.
- [16] P. Neugebauer, “Geometrical cloning of 3d objects via simultaneous registration of multiple range images,” in *Proc. Int. Conf. On Proc. Int. Conf. on Shape Modeling and Application*, Mar 1997, pp. 130–139.
- [17] A.E. Johnson and S. Kang, “Registration and integration of textured 3-d data,” in *Proc. of Int. Conf. on 3D Digital Imaging and Modeling*, May 1997, pp. 234–241.
- [18] K. Nishino and K. Ikeuchi, “Robust simultaneous registration of multiple range images,” in *Proc. of the 5th Asian Conf. on Computer Vision*, January 2002, vol. 2, pp. 455–461.
- [19] Mark D. Wheeler, *Automatic Modeling and Localization for Object Recognition*, Ph.D. thesis, School of Computer Science, Carnegie Mellon University, 1996.
- [20] E. Polak, *Computational Methods in Optimization*, New York: Academic Press, 1971.
- [21] David A.H. Jacobs, *The State of the Art in Numerical Analysis*, London: Academic Press, 1977.
- [22] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis*, New York: Springer-Verlag, 1980.

- [23] J. L. Bentley J. H. Friedman and R. A. Finkel, “An algorithm for finding best matches in logarithmic expected time,” *ACM Transactions on Mathematical Software*, vol. 3, no. 3, September 1977.
- [24] J.L. Bentley, “K-d trees for semidynamic point sets,” in *Proc. of the 6th Annual Symposium on Computational Geometry*, 1990, pp. 187–197.