

# 非線形最適化計算による 画像列からの3次元形状復元の研究

(課題番号 16500118)

平成16年度～平成18年度  
科学研究費補助金(基盤研究(C)(2))

研究成果報告書

平成19年3月

研究代表者：浅田 尚紀  
(広島市立大学 学長・情報科学部 教授)

## はしがき

物体を移動カメラで撮影した画像列から物体形状とカメラ運動を同時に復元する「運動からの形状復元」はコンピュータビジョンの基本問題であると同時に、実物体の3次元モデル化や複合現実への利用など様々な応用が期待される技術である。従来は線形定式化による解法(例えば因子分解法)が研究の中心であったが、最近では線形近似の必要がない非線形最適化による解法が注目されている。

我々は、実問題に忠実に適用でき拡張性の高い非線形最適化手法に着目し、実写画像列からの3次元形状復元のための高速で実用的なアルゴリズムの開発を行った。その研究成果について、ここに報告する。

## 研究組織

研究代表者 浅田 尚紀 (広島市立大学 学長・情報科学部 教授)  
研究分担者 椋木 雅之 (広島市立大学 情報科学部 助教授)  
研究分担者 青山 正人 (広島市立大学 情報科学部 助手)  
研究分担者 馬場 雅志 (広島市立大学 情報科学部 助手)

## 交付決定額 (配分額)

	直接経費	間接経費	合計
平成 16 年度	1,200,000 円	0 円	1,200,000 円
平成 17 年度	1,200,000 円	0 円	1,200,000 円
平成 18 年度	1,000,000 円	0 円	1,000,000 円
総 計	3,400,000 円	0 円	3,400,000 円

## 研究発表

### (1) 学会誌等

1. Naoki Asada, Masayuki Mukunoki, Tsuyoshi Migita, Masahito Aoyama :  
Large Object Shape Recovery from Uncalibrated Camera Motion by Non-Linear Optimization,  
Proc. of International Conference on Signal and Image Processing, pp.151-156, 2004-08.
2. 椋木 雅之, 右田 剛史, 青山 正人, 浅田 尚紀 :  
非線形最適化による建物画像列からの全周形状一括復元のための初期値設定法,  
情報処理学会論文誌, Vol.45, No.SIG13(CVIM), pp.64-73, 2004-12.
3. Masayuki Mukunoki, Kazutaka Yasuda, Naoki Asada :  
3D Model Generation from Image Sequences using Global Geometric Constraint,  
Advances in Visual Computing – International Symposium on Visual Computing  
2005 (LNCS3804), pp.470-477,2005-12.
4. 保田 和隆, 椋木 雅之, 浅田 尚紀 :  
大局的幾何学制約を用いた特徴点追跡に基づく画像列からの3次元モデル生成,  
情報処理学会 CVIM 論文誌, SIG10(CVIM15) vol.47, pp. 96-106, 2006-07.

### (2) 口頭発表

1. 右田 剛史, 浅田 尚紀 :  
画像列からの3次元形状復元における非線形最小2乗法の高速化と安定化に関する研究,  
情報処理学会研究報告, Vol.2004, No.40, 2004-CVIM-144(21), pp.157-172, 2004-05.
2. 椋木 雅之, 右田 剛史, 青山 正人, 浅田 尚紀 :  
非線形最適化による画像列からの全周形状復元のための初期値設定の検討,  
画像の認識・理解シンポジウム (MIRU2004) 論文集, Vol.I, pp.745-750, 2004-07.
3. 保田 和隆, 椋木 雅之, 浅田 尚紀 :  
大局的幾何学制約を用いた特徴点追跡に基づく画像列からの3次元モデル生成,  
画像の認識・理解シンポジウム (MIRU2005) 論文集, pp.159-166, 2005-07.
4. 灰垣 雅崇, 椋木 雅之, 浅田 尚紀 :  
モザイクングを用いた疑似広角撮影画像列からの形状復元,  
平成17年度電気・情報関連学会中国支部連合大会, 22-30, pp.403-404, 2005-10.
5. 藤村 一之, 椋木 雅之, 浅田 尚紀 :  
線分要素に基づく画像列からの3次元形状モデルの生成,  
情報科学技術レターズ (FIT2006), LI-012, pp.205-206, 2006-09.

(3) 出版物

なし

工業所有権等の取得

なし





# 目次

<b>第1章</b>	<b>はじめに</b>	<b>1</b>
1.	研究目的 . . . . .	2
1.1	目的 . . . . .	2
1.2	学術的特色, 独創性, 意義 . . . . .	2
1.3	研究の位置づけ . . . . .	3
2.	研究成果の概要 . . . . .	3
2.1	非線型最適化計算の高速化, 安定化 . . . . .	3
2.2	画像特徴の高精度追跡 . . . . .	4
<b>第2章</b>	<b>非線型最適化計算の高速化, 安定化</b>	<b>7</b>
1.	非線形最適化による建物画像列からの全周形状一括復元のための初期値設定法	8
2.	Large Object Shape Recovery from Uncalibrated Camera Motion by Non-Linear Optimization . . . . .	20
3.	モザイクングを用いた疑似広角撮影画像列からの形状復元の評価 . . . . .	27
4.	線分要素に基づく画像列からの物体形状モデルの生成 . . . . .	40
5.	陰影情報と特徴点对応の統合による画像列からの3次元形状復元 . . . . .	55
<b>第3章</b>	<b>画像特徴の高精度追跡</b>	<b>71</b>
1.	大局的幾何学制約を用いた特徴点追跡に基づく画像列からの3次元モデル生成	72
2.	3D Model Generation from Image Sequences using Global Geometric Constraint . . . . .	86
3.	奥行き制約による対応点探索を用いた画像列からの3次元モデル生成 . . . . .	92
4.	線分要素の自動対応付けに基づいた画像列からの疎な3次元形状復元 . . . . .	108

# 第1章

## はじめに

# 1 研究目的

## 1.1 目的

物体を移動カメラで撮影した画像列から物体形状とカメラ運動を同時に復元する「運動からの形状復元」はコンピュータビジョンの基本問題であると同時に、実物体の3次元モデル化や複合現実への利用など様々な応用が期待される技術である。従来は線形定式化による解法(例えば因子分解法)が研究の中心であったが、最近では線形近似の必要がない非線形最適化による解法が注目されている。

我々は、実問題に忠実に適用でき拡張性の高い非線形最適化手法に着目し、実写画像列からの3次元形状復元のための高速で実用的なアルゴリズムの開発を行う。具体的には、下記の技術課題について検討を行う。

- 非線形最適化計算の高速化と安定化

従来の3次元形状復元では、一般的な非線形最適化アルゴリズム(Levenberg-Marquardt法、共役勾配法、最急降下法など)が用いられてきたが、我々は3次元形状復元問題特有の性質に着目することによって、Levenberg-Marquardt法と共役勾配法の長所を兼ね備えた「前処理つき共役勾配法」を開発している。このアルゴリズムは、特徴点が数千点規模の問題でLevenberg-Marquardt法の約10倍の高速化を実現している。本研究では、メモリ容量の点でも有利な前処理つき共役勾配法をさらに改良し、大規模問題における実用的な非線形最適化アルゴリズムを開発する。

また、一般に非線形最適化問題には多数の局所解が存在し、解探索が不安定となる欠点がある。この問題に対して、我々は3次元形状復元問題の局所解の性質を明らかにするとともに、適切な初期値の設定法について検討する。

- 画像特徴の高精度追跡

高品質な3次元形状復元を行うには、実写画像列から点や線などの特徴を高精度に追跡する必要がある。従来は人手あるいは半自動の入力を行っていたが、大規模な対象を高密度に復元するには数万から数百万の特徴を検出し画像間で対応付ける必要があり、実用化のためには特徴追跡の自動化が必要不可欠である。我々は、アフィンマッチングによる大局的な制約下での局所的なテンプレートマッチングによる高精度、高密度な特徴点追跡のアルゴリズムを開発している。本研究では、このアルゴリズムを改良し、膨大な画像データからの点および線の自動特徴追跡アルゴリズムを開発する。

## 1.2 学術的特色, 独創性, 意義

コンピュータビジョンの多くの問題は、対象モデルと観測情報との誤差の2乗和を最小化するという非線形最適化問題として定式化することが可能である。これは、不完全で不正確な画像情報から、全体として整合性のある結果を得るための非常に強力な手法である。

しかし、従来は計算アルゴリズムについての工夫や高速化についての議論はほとんどなされず、古典的なアルゴリズムがそのまま用いられていた。我々は、3次元形状復元という問題に特有の性質を利用することによって、従来手法の数倍から数十倍の高速化が可能なアルゴリズムを独自に開発している。本研究によってアルゴリズムをさらに改良し、大規模問題への適用可能性を実証することができれば、実写画像からの自動実時間3次元形状復元に向けて大きく前進することが期待できる。

また、3次元形状復元結果の精度と品質を大きく左右する画像間の特徴対応では、従来は局所対応の組み合わせを投票によって評価し大局的整合性を求める手法が主流であったのに対し、本研究では最初に画像そのものを対応させることで大局的制約を与えるトップダウン手法を採用する。この方法は、明らかな誤対応の確実な排除や計算コストの削減などの効果が期待できるという特徴がある。

### 1.3 研究の位置づけ

小型物体に対する画像からの3次元形状復元については多くの研究がなされているが、大規模な実問題を扱った研究は数少ない。いくつかの研究グループが建築物等の大型対象物の形状復元や表示技術の研究を行っているが、物体の3次元計測に特殊なレーザーレンジセンサを用いた方法や、非線形最適化アルゴリズムに従来手法であるLevenberg-Marquardt法を用いたものであり、その計算コストは高いものとなっている。

これに対して、本研究では、高密度かつ高精度な特徴点検出、非線形最適化アルゴリズムの高速化と安定化、という3次元形状復元の最も重要で本質的な課題に取り組むものであり、その研究成果はコンピュータビジョン研究の進展と実用化に大きく貢献するものである。

## 2 研究成果の概要

### 2.1 非線形最適化計算の高速化、安定化

#### 1. 非線形最適化による建物画像列からの全周形状一括復元のための初期値設定法

画像列からの形状復元問題は、一般に非線形最適化問題となるが、非線形最適化では、初期値設定により局所解におちいり、正解が得られないことが問題とされている。本研究では、屋外建物の全周囲形状復元問題に対して、簡便な初期値設定を行い、非線形最適化により直接解を求める方法を開発した。この初期値は、全周囲復元の問題設定を反映したものであり、実場面において比較的広く適用できる。また、局所解におちいった場合に、解を修正することにより正しい形状復元を行う手法も開発した。画像数数百枚、特徴点数数百点のシミュレーションおよび実画像を用いた実験を行い、有効性を示した。

#### 2. モザイクングを用いた疑似広角撮影画像列からの形状復元の評価

同一カメラ位置から、カメラの姿勢を変えながら複数の画像を撮影する手法を疑似広角撮影と呼ぶ。非線形最適化を用いた画像列からの形状復元では、それぞれの画像内に多くの特徴点が観測されている程、安定した形状復元が行える。疑似広角撮影では、同一カメラ位置から観測できる特徴点数が増加することにより、擬似的に広角レンズを用いて撮影したのと同様な効果が得られ、形状復元の安定化がはかれると考えられる。本研究では、疑似広角撮影した画像をモザイクングにより1枚の画像に合成することで、実際に形状復元が安定化することを、シミュレーションおよび実画像を用いた実験により、定量的に示した。

### 3. 線分要素に基づく画像列からの物体形状モデルの生成

画像列からの形状復元では、「点」を単位として処理が行われることが多い。しかし、物体の直線状の輪郭上では「点」を単位とした対応付けには本質的に曖昧さが含まれる。また、形状復元に、より大きな構造的単位を導入すれば、物体形状を直接的に表現することができるようになり、形状復元の安定化につながる。そこで、本研究では、「線」を単位とした画像列からの形状復元手法を提案する。提案手法では、非線形最適化手法において、特徴点に対して線分方向に曖昧性を持たせることで、「線」を表現する拡張を行った。提案手法により、物体輪郭の線分を単位として形状復元が行えることを、シミュレーション画像および実画像実験により示した。

### 4. 陰影情報と特徴点对応の統合による画像列からの3次元形状復元

特徴点对応を用いた画像列からの3次元形状復元では、検出された特徴点の3次元座標のみしか得られないため、物体表面の細かな形状まで表現できないという欠点がある。これに対して、陰影情報を利用した形状復元手法では、観測した画像の画素毎の法線情報が得られるため、そこから細かな物体形状を復元できるが、法線から3次元形状を得る処理が適切に行えないという問題がある。そこで、本研究では、特徴点对応により得られた特徴点の3次元座標と、陰影情報より得られた法線を統合することで、対象物体の詳細な3次元形状を得る手法を実装し、シミュレーションおよび実画像を用いた実験により、その安定性について評価した。

## 2.2 画像特徴の高精度追跡

### 1. 大局的幾何学制約を用いた特徴点追跡に基づく画像列からの3次元モデル生成

非線形最適化計算による実写画像列からの3次元形状復元問題に関して、位置・姿勢が未知のカメラを用いて物体の全周囲を撮影した画像列から、3次元モデルを生成する一連の手順を構成した。特に、画像特徴追跡においてアフィン制約と奥行き制約という2種類の幾何学的制約を導入することによって高品質な3次元モデルを自動生成するアルゴリズムを実現した。

- (a) 視差のある2画像間の変形をアフィン変換で近似し、対応する画素の近傍領域内で特徴点を探索するというアフィン制約により、初期的な特徴点追跡を行う。

- (b) 非線形最適化計算により，特徴点から物体特徴点の3次元座標とカメラの位置・姿勢を推定する．
- (c) カメラの位置・姿勢情報を用いて，物体特徴点の3次元座標を各画像に再投影し，画像上の特徴点座標との誤差が大きな特徴点を除去する．
- (d) 特徴点から復元した物体特徴点の3次元座標とカメラ位置・姿勢から，物体表面を覆うドロネー3角面を生成し，特徴点の奥行き方向の存在範囲を限定する．
- (e) 2画像間のエピポーラ線上の奥行き制約の範囲内で対応する特徴点を探索し追加する．
- (f) 上記の2から5を特徴点数がほぼ一定に収束するまで繰り返す．
- (g) 得られた3次元形状の各3角面にテクスチャ画像を付与することにより3次元モデルを生成する．

本手法では，大きな誤対応点を除去し信頼性の高い対応点を追加するという処理を繰り返すことによって，シミュレーション実験では36枚の画像から11,277特徴点，実画像実験では25枚の画像から15,751特徴点という高密度な3次元モデルを自動生成できることを確認した．

## 2. 奥行き制約による対応点探索を用いた画像列からの3次元モデル生成

上記の大局的幾何学制約を用いた3次元モデル生成手法では，物体の奥行きを推定するために，3次元特徴点の円筒面への投影と2次元ドロネー法を利用していた．この方法は，高速であるが，物体形状が円筒断面に対して凹状の場合，適切に奥行き推定が行えない．本研究では，3次元特徴点に対してマーチンキューブ法を適用して奥行き推定することにより，任意の形状に対して特徴点を高精度に追跡し，3次元モデルを生成する手法を提案した．複数の実画像に対して，本手法を適用し，3次元モデルが適切に生成できることを示した．

## 3. 線分要素の自動対応付けに基づいた画像列からの疎な3次元形状復元

これまでの研究では，画像列から多数の特徴点を抽出し，追跡することで高精度な3次元モデルを生成するアプローチをとっていた．しかし，復元対象となる物体が，ビルや電車など直線状の明瞭な輪郭をもつ物体（人工物）の場合，物体の頂点に対応する特徴点を確実に抽出，追跡できれば，高精度な3次元モデルを生成することができる．本研究では，対象物体が上記のような人工物であることを仮定し，画像内の線分要素の交点が物体の頂点であることに着目して，少数の安定した特徴点を抽出，追跡して，3次元形状を復元する手法を提案した．実験では，シミュレーション画像において，提案手法が適切に動作することを示した．





## 第2章

### 非線型最適化計算の高速化，安定化

- 非線形最適化による建物画像列からの全周形状一括復元のための初期値設定法
- Large Object Shape Recovery from Uncalibrated Camera Motion by Non-Linear Optimization
- モザイクングを用いた疑似広角撮影画像列からの形状復元の評価
- 線分要素に基づく画像列からの物体形状モデルの生成
- 陰影情報と特徴点对応の統合による画像列からの3次元形状復元

# 非線形最適化による建物画像列からの 全周形状一括復元のための初期値設定法

## Initialization Method for Batch Structure from Motion for Entire 3D Shape Recovery from a Building Image Sequence by Non-linear Optimization

### あらまし

本論文では、建物のような屋外の大型の対象物を撮影した画像列から、対象物の全周形状を復元する問題を扱う。画像列からの形状復元は、一般に非線形最適化問題となるが、非線形最適化計算では、初期値設定により局所解におちいり、正解が得られないことが問題とされていた。それに対し、本論文では非線形最適化計算により直接解を求めるための、簡便な初期値設定法を提案する。この初期値設定は、全周復元の問題設定を反映したものであり、実場面において比較的広く適用できる。また、局所解におちいった場合に、解を修正することにより正しい形状復元を行う手法も提案する。最後に、画像数 30~90 枚、特徴点数 300~1200 個規模のシミュレーションおよび実画像を用いた実験によって、本手法の有効性を示す。

### Abstract

We propose a method of setting the initial values of shape and camera parameters for structure from motion based on non-linear optimization method. The initial values of the parameters in the optimization process is crucial in solving the problem. Thus the direct application of the non-linear optimization process has not been fully used in shape recovery. In this paper, first, we investigate an initialization method, which is very simple and relatively generic to the entire shape recovery problem. Then, we propose a method for restoring the optimization process from a local minimum. Experimental results using synthetic and real images have shown that our algorithm stably recover 3D shapes.

## 1 はじめに

物体の三次元形状を取得することは、CG モデルの生成や物体測定、3次元デジタルアーカイブなど多くの分野で必要とされている。小型の物体に対しては、レーザレンジセンサやステレオ画像計測器など、比較的簡便な装置が開発されており、形状取得が容易になりつつある。しかし、屋外の建物の全周形状のように、計測環境を制御できない場所にある大型物体に対しては、多くの人手をかけたり、大がかりな装置を用いた測量が必要であった。このような問題に対し、物体を撮影した画像列からその物体の3次元形状を復元する、画像列からの形状復元の手法は、大がかりな装置なしに形状復元が行

えるため、盛んに研究されている。

画像列からの形状復元では、物体の形状は、物体上に設定した特徴点群によって表現される。形状復元問題は、各画像上の特徴点の2次元座標から、その特徴点の3次元座標と撮影したカメラの位置・姿勢を同時に求める問題として定式化できる。その解法の基本的な原理は、「画像上の特徴点と、その特徴点の推定した3次元座標を画像上に再投影した点との誤差が最小となるようパラメータ推定を行う」という再投影誤差最小化の考えに基づいている。

再投影誤差最小化は、正規分布の誤差を仮定した下での、最尤推定の考えに従った定式化である [1]。カメラの投影モデルとして透視投影

を仮定すると、この再投影誤差最小化は、非線形最適化問題となる [2]。この非線形最適化問題を非線形最適化計算により直接解けば、最尤推定の意味で最適性が保証されるため、高精度な解が得られる。また、最適化の目的関数は、再投影誤差最小化の考えをそのまま利用したものとなる。その意味で、非線形最適化計算による解法は、単純かつ有効な方法といえる。

このような非線形解法は、バンドル調整 (bundle adjustment) と呼ばれ広く利用されている。バンドル調整は従来、他の手法で得られた復元結果の精度を向上するための“補正処理”として主に利用されてきた [1, 3]。これは、この非線形解法には高精度な形状復元を行えるという利点がある一方、以下の問題があるためと考えられる。

- 解析的な解法がなく、繰り返し処理による解の反復改良が必要なため、処理速度が遅い。
- 反復改良の初期値の与え方により、求めたい正解以外の局所解が得られる場合がある。

このうち、処理速度については、高速化を目指した研究がなされており [4, 5, 6]、計算機性能の向上と相まって、現状ではそれほど問題にならなくなってきている。局所解については、対象とする問題のクラスを限定し、適切な初期値設定を行うことで回避可能であると考えられる。

そこで、本稿では、全周形状を画像列から復元するというクラスの問題に対し、非線形最適化を直接適用する際の簡便な初期値設定方法を提案し、その適用可能範囲について検討する。そのために、全特徴点のうち、画像列中の画像 1 枚に現れる点の割合を表す平均出現率に着目する。この値が一定以上であれば、ほとんどの場合、設定した初期値から正解に収束することを実験により示す。また、局所解に陥った場合に、解の修正操作により正解を得るための手法についても提案する。これにより、他の手法で得られた結果に対する“補正処理”と位置付けられていたバンドル調整 (非線形最適化計算に

よる解法) が、全周復元の問題に対しては有効な解法であることを示す。

以下、2. では、本稿で扱う全周形状の復元問題と形状復元に関する従来手法について述べる。3. では、非線形最適化計算による解法について説明する。4. では本稿で提案する全周復元のための簡便な初期値設定法と、局所解に陥った場合の解の修正法について述べる。5. では、提案した初期値設定法の適用可能範囲について、実験を行う。6. では結論と今後の課題について述べる。

## 2 画像列からの全周形状復元

### 2.1 全周復元問題

本稿では特に、対象物体である建物の近くから周囲 360 度を撮影した画像列を用いて、形状を一括復元する問題を扱う。図 1 に示すように、撮影モデルとして、「円周撮影」と「疑似広角撮影」の 2 つを導入する。図 1(a) の「円周撮影」は、各カメラ位置で 1 枚の画像を撮影しながら円周上を一方向に移動するものである。一方、図 1(b) の「疑似広角撮影」は、円周撮影と同様の移動をしながら、各カメラ位置でカメラ方向を変化させて複数の撮影を行うものであり、より広い画角範囲の情報を得ることができる。疑似広角撮影の有効性は、実験の節で示す。

実際の撮影では、正確な円周上を移動するのではなく、建物の方向を向きながら全周を撮影するが、そのような場合も含めて「円周撮影」「疑似広角撮影」と呼ぶものとする。

実在する建物の撮影では、建物自身による隠蔽のため、1 枚の画像からは最大でも全周の半分程度の情報しか得られない。更に、建物周辺に木立や他の建造物が存在するため、遠方から撮影できず、建物に近接した位置から撮影した画像列を使うことが多い。そのため、全特徴点のうち 1 枚の画像内に現れる点の数は少なくなる。画像内に現れない特徴点のことを隠れ点とよぶ。本論文で扱う全周復元問題は、隠れ点が多い問題設定である。このような場合、復元

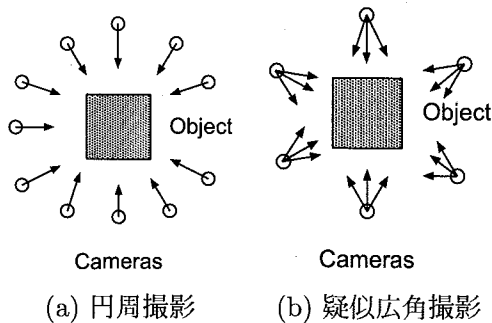


図 1: 撮影モデル

すべき情報に対して、利用できる情報が減るため、一般に復元が困難となる。

## 2.2 従来研究

画像列からの形状復元については、従来から多数の研究がなされている。

多数の画像を一括して扱い形状復元する手法として、因子分解法 [7] がある。因子分解法は、観測した特徴点をならべた計測行列が、特徴点の 3 次元位置成分とカメラの位置・姿勢成分に分離可能なことを利用している。以下の 2 つの制約が満たされている場合、特異値分解を用いて、一意の解を高速に求めることができる。

(a) 全特徴点が全画像で観測されている。  
 (b) カメラの投影がアフィン投影で近似できる。  
 再投影誤差最小化の観点からみると、この手法は、上記の制約のもとでの近似解法とみなすことができる。カメラと物体間の距離が十分離れている場合、この制約は満たされるが、建物の全周復元では一般にこれらの制約は満たされない。これらの制約を緩和する事によって、より広範な画像列を対象に、安定な形状復元を目指す研究も多数行われている [8, 9, 10, 11, 12] が、隠れ点の多さや射影歪みの強さにより十分な精度が得られない、隠れ点と射影歪みに同時に対処することが難しいなど、全周復元への適用には不向きである。

別の方法として、数枚の画像からの形状復元結果を繋ぎ合わせて近似解を得て、それを初期値としてバンドル調整 (非線形最適化) により、

再投影誤差を最小化する方法 [3] もある。2 または 3 枚の画像からの形状復元では、カメラの位置・姿勢の相対関係を線形化可能であり、探索を要しない閉じた形式の解が与えられる [13, 14]。これらの結果を全周に渡って統合し、非線形最適化の初期値とすることにより、一意の解が得られる線形解法の利点と、高精度な解が得られる非線形手法の利点を兼ね備えた復元が行えることから、大規模な画像列を用いた形状復元を行う場合の 1 つの現実的方法といえる。線形化を行わない場合でも、画像を逐次的に増やしながらか非線形解法を適用することにより、初期値推定の問題を解決する試みがなされている [4, 15]。しかし、部分画像列からの形状復元は全画像を用いた復元に比べ、解の精度が低くなる傾向がある [16]。特に 2 枚の画像を使う場合、射影幾何学に基づく代数的操作が誤差を増幅することも指摘されている [17]。さらに、多段階の手法では、手順も複雑となり、データ管理が繁雑になる上、誤差の蓄積により、適切な初期値推定が行えないことも考えられる。本論文では、全周復元を仮定した場合、このような複雑な初期値設定は必ずしも必要ないことを実験により示す。

本論文と同様に、再投影誤差の最小化を非線形最適化計算により直接解く方法もいくつか報告されている。文献 [2] は、近似解も要さず観測データのみから初期値を構成する方法によって非線形最適化により形状復元が行えることを示した。しかし、全周形状の一括復元は対象としておらず、また、具体的な問題での初期値の適用可能性についても、十分に検討されていない。

## 3 再投影誤差最小化による形状復元

### 3.1 定式化

物体は  $P$  個の特徴点で表現される。各特徴点は、物体とカメラに共通のある適当な座標系 (世界座標系) 上にあり、固有の 3 次元座標  $s_p$  ( $p = 1 \dots P$ ) を持つ。これらの点は最大  $F$

枚の画像上に投影され観測される。第  $f$  画像 ( $f = 1 \dots F$ ) 上で観測された、第  $p$  特徴点の座標を  $u_{fp}$  とする。特徴点  $p$  が画像  $f$  で観測されていない場合  $u_{fp}$  は定義されない。

各画像を撮影したカメラの位置・姿勢は、世界座標系における3次元の回転と並進 ( $R_f, t_f$ ) で表される。カメラ位置は直接のパラメータではなく、 $-R_f^T t_f$  の形で与えられる。投影モデルは透視投影とし、焦点距離は全画像を通じて共通の既知の値  $l$  とする。

これらのパラメータを用いて、形状復元問題は次のように定式化される [2]。

$$\begin{aligned} \arg \min_{\mathbf{x}} E(\mathbf{x}) & \quad (1) \\ E(\mathbf{x}) & := \sum_{(f,p)} |\mathcal{P}[R_f \mathbf{s}_p + \mathbf{t}_f] - \mathbf{u}_{fp}|^2 \\ \mathbf{x} & := (\mathbf{s}_p, \mathbf{q}_f, \mathbf{t}_f \text{を並べたベクトル}) \\ \text{ただし, } \mathcal{P}(x, y, z)^T & := \frac{l}{z}(x, y)^T \end{aligned}$$

ここで  $R_f$  は自由度3の回転行列であり、 $\mathbf{q}_f$  は  $R_f$  のパラメータを並べたベクトルである。また、特徴点の画像座標とその画像間での対応  $u_{fp}$  は予め与えられているものとする。

この式は、推定した形状  $\mathbf{s}_p$  を推定したカメラパラメータ  $R_f, \mathbf{t}_f$  で投影した場合の投影座標  $\mathcal{P}[\cdot]$  と、観測値  $u_{fp}$  との誤差（これを再投影誤差と呼ぶ）の2次ノルムの総和が最小となることを要求するものであり、観測座標に等方性正規分布の誤差を仮定した場合の最尤推定を与える式である。この式を最小化することにより、最尤推定の意味で最適な形状復元が行えることになる。

各特徴点は、2枚以上の画像で観測されていないければ、その位置を推定することができない。各カメラの位置・姿勢を求めるには、2画像以上で対応する特徴点が6点以上、または、3画像以上で対応する特徴点が4点以上観測されている必要がある。信頼できる推定のためには、これらの下限値よりも多くの画像で多くの特徴点が観測される必要がある。

## 3.2 最適化アルゴリズム

全周復元の問題に対して、式(1)を目的関数とする非線形最適化計算により、解を求める。一般に、非線形最適化では、繰り返し処理により解を逐次更新する反復法が用いられる。計算法としては、Levenberg-Marquardt法などの任意の汎用非線形最適化手法が利用できるが、問題の規模が大きい場合、より計算量の少ない、前処理付き共役勾配法 [4, 5] が適している。

前処理付き共役勾配法は、以下の式で表される。

$$\begin{aligned} \mathbf{x}_0 & = \text{初期値} \\ \mathbf{d}_0 & = C^{-1} \mathbf{g}_0 \\ \mathbf{x}_{i+1} & = \mathbf{x}_i - \alpha_i \mathbf{d}_i \\ \mathbf{d}_{i+1} & = C^{-1} \mathbf{g}_{i+1} + \beta_i \mathbf{d}_i \\ \text{ただし } \alpha_i & = \arg \min_{\alpha} E(\mathbf{x}_i - \alpha \mathbf{d}_i) \\ \beta_i & = \frac{(C^{-1} \mathbf{g}_{i+1})^T (\mathbf{g}_{i+1} - \mathbf{g}_i)}{\mathbf{g}_i^T C^{-1} \mathbf{g}_i} \\ \mathbf{g}_i & = \nabla E(\mathbf{x}_i) \end{aligned}$$

ここで、 $C$  は前処理行列であり、ヘッセ行列 ( $\partial^2 E / \partial x_i \partial x_j$ ) の対角ブロック成分のみを残した帯行列を用い、16反復毎に再計算を行う。本稿で扱った問題では、画像数  $F = 30 \sim 90$ 、特徴点数  $P = 100 \sim 1200$ 、未知数が  $1000 \sim 5000$  程度となるが、この場合、計算時間は PentiumIII 1GHz の PC を用いて 0.1~10 秒程度となる。

## 4 初期値と収束可能性

### 4.1 注視回転による初期値

建物画像列からの全周復元では、カメラはほぼ水平な平面上に位置し、建物方向を向いて移動しながら撮影が行われると考えられる。また、カメラは建物に対して、原則として1方向に移動しており、画像列は撮影順序に従って与えられるものとする。この撮影状況を物体上に設定した注目点を中心にカメラが等速円運動をしながら物体を観測する注視回転によりモデル

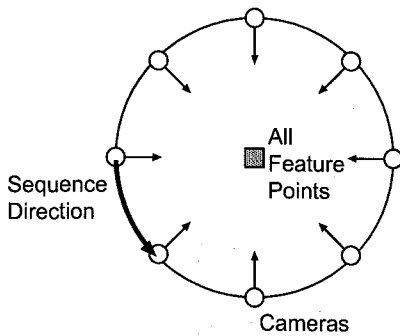


図 2: 注視回転による初期値

化する(図 2)。ここで、カメラ位置と光軸は全て同一平面(水平面)上にあるものとする。

初期値として必須の成分には、カメラの位置、姿勢情報と、復元対象の形状情報の 3 種類がある。このうち、形状情報は一般に未知であり、初期値として何らかの仮定を導入することは現実的でない。そこで、形状の初期値としては全特徴点が原点に位置するという縮退した状態を用いる。カメラの位置、姿勢情報については、注視回転を仮定して設定する。この初期値(以下、注視回転による初期値と呼ぶ)を数式で表すと次式のようになる。

$$\begin{cases} R_f = R_{\omega f}^y & \mathbf{t}_f = (0, 0, r)^T & \text{for all } f \\ \mathbf{s}_p = (0, 0, 0) & & \text{for all } p \end{cases} \quad (2)$$

ここで、 $R_{\omega f}^y$  は、 $y$  軸(図 2 の紙面鉛直方向)回りの角度  $\omega f$  の回転を表す行列とする。また、 $r$  はカメラの円軌道の半径であるが、計算空間の 3 次元座標系には物理単位が与えられていないため、 $r = 1$  として良い。従って、撮影状況に応じて与えるべきパラメータは、全回転量  $\omega F$  (または、画像間の平均回転移動量  $\omega$ ) のみとなる。全周復元では、撮影時のカメラ軌跡の方向( $\omega F$  の符号)を考慮して、 $\omega F = 360$  度または  $\omega F = -360$  度のいずれかを選ぶ。

このような初期値設定により、どの範囲の問題を解くことができるかが重要であるが、非線形性のため、解析的な検討は困難である。経験的に得られた傾向を以下に列挙する。

- 対象物体の形状にはそれほど依存せず、1 点に縮退した初期値から正解を得ること

ができる。

- 対象物体に近接して撮影するほど局所解に陥りやすく、復元が失敗する可能性が高くなる。
- 図 1(d) の疑似広角撮影では、各撮影点で仰角や方位角が大幅に変動するが、これに対しても、式 (2) を初期値として正解が得られる。また、疑似広角撮影の方が円周撮影よりも安定して解が得られる。
- 全回転量の大きさにもそれほど影響を受けない。実際の回転量が 270 度であっても 360 度を初期値として正解がえられる。
- カメラ軌跡の方向は重要で、撮影時と異なる設定をすると、局所解に陥る可能性が高い。

5. の実験において、上記の点についてシミュレーションによる評価を行う。

## 4.2 平均出現率

ある与えられた形状復元問題が非線形最適化計算により解けるかどうかは、初期値の設定法だけでなく、カメラ-物体間距離や、物体形状の凹凸、利用する画像数、特徴点数など様々な要因に依存する。これらを総合的に表す指標として、特徴点の平均出現率を用いる。平均出現率は、各特徴点が全画像のうち何%で観測されるかを表す量で、文献 [7] で用いられている「観測行列の fill-ratio」と等価なものである。各特徴点が平均して全画像の  $r\%$  で観測される場合、平均出現率は  $r\%$  とする。これは、各画像に平均して全特徴点のうち  $r\%$  が観測されると考えることもできる。すなわち、平均出現率は、画像間での特徴点の対応、あるいは観測された点の割合を表している。

カメラの画角が一定のままカメラ-物体間距離が縮まれば、画像内の特徴点数は減少し、平均出現率は低下する。同様に、物体形状の凹凸が多ければ、隠れ点を多く生じ、平均出現率は低下する。このように、平均出現率は、与えら

れた2次元情報からの形状復元の困難さを反映すると考えられる。

### 4.3 解の修正

注視回転による初期値は、多くの場合で正解を得ることができる。しかし、隠れ点が多量の画像列の場合には、局所解に陥ることがある。このような場合にも、式(1)の枠組の中で、特徴点の削除と再追加を繰り返すことにより、解を修正して、局所解から抜け出すことができる場合がある。そのために以下の手順をとる。

- (i) 少数の画像のみに現れる特徴点を除去する。
- (ii) 残った特徴点のみを用いて、形状と運動を再推定する。
- (iii) 除去した特徴点を再追加する。

以下、この手順について説明する。

式(1)の目的関数において総和の範囲を限定した次式を導入する。

$$E(\mathbf{x}, L) = \sum_{(f,p) \in S(L)} |P[R_f s_p + \mathbf{t}_f] - \mathbf{u}_{fp}|^2$$

ここで、 $S(L)$ は考慮すべき添字の集合で、 $L$ は制御パラメータである。特徴点 $p$ が $L$ 枚以上の画像に現れる場合のみ $S(L)$ に $(f,p)$ を含めるものとする。

理論上は、特徴点は2画像以上で観測されていれば( $L=2$ )復元可能であるが、少数の画像でしか現れない特徴点が誤った位置に推定されることによって、カメラ位置や姿勢が誤って推定され、局所解となる場合が多い。手順(i)では、 $L$ を増加させることによって、局所解の原因となっているこのような特徴点(群)を一旦除去する。手順(ii)では、残った点を用いて、形状復元を行うことにより、カメラ位置、姿勢を推定しなおす。多数の画像で観測されている特徴点のみを用いると、カメラ位置、姿勢や形状を正しく推定できる傾向がある。手順(iii)では、手順(ii)で推定された正しい値をもとにして、除去した特徴点群を含めた全ての特徴点を用いて式(1)を最小化する。これによって、局所解から脱出する。一回で解を修正しきれない場合、 $L$ の増加、減少を複数回繰り返すことで修正を行う。

一般に、局所解では、 $E(\mathbf{x}, L)$ の値が大きいものに対して、局所解の原因となっている特徴点が除去できると、 $E(\mathbf{x}, L)$ の値が急激に減少する。 $L$ の増加、減少は、 $E(\mathbf{x}, L)$ の値を参照しながら、基本的に手動により行うが、以下のアルゴリズムによって、自動的に決定することも考えられる。評価実験においては、下記のパラメータ $L$ の自動設定法についても評価する。

#### [パラメータ $L$ の自動設定法]

・最適化処理の中で一定の繰り返し回数毎に以下の判定を行う

- (a)  $L > 2$ であれば、 $L$ を1減らす。
- (b)  $L = 2$ であれば、 $E(\mathbf{x}, L')/E(\mathbf{x}, 2) > T (L' = 3, \dots, 9, T$ は閾値)となる最初の $L'$ に $L$ を更新する。条件を満たすものがなければ、 $L = 2$ のままとする。

解の修正では、局所解に陥った際に、 $L$ を増やすことによって平均出現率の高い特徴点のみからなる部分問題を解いている。この部分問題は元の問題よりも容易な問題となっているため、正解が得られる可能性が高い。その結果を利用するため、局所解から抜け出すことができると解釈できる。この手順のみを用いて必ずしも局所解から抜け出せるわけではないが、パラメータ $L$ を増加させても十分な数の特徴点が残る画像列では、正解が得られると期待できる。

## 5 実験

本節では、全周復元問題に対して、提案した初期値設定がどの程度有効であるか確かめるため、数値的に生成した疑似建造物を用いた模擬実験、並びに、実画像での実験結果を示す。

### 5.1 シミュレーション画像による復元

凹凸のある建物を想定し、球を変形させた次式の曲面を対象物体とする。ただし、 $x-z$ 平面を地面と考える。

$$D \begin{bmatrix} [1 - \nu + \mu \cos(\rho(\varphi - \pi/4))] \cdot \cos \theta \cos \varphi \\ [1 - \tau] \cdot \sin \theta \\ [1 - \nu + \mu \cos(\rho(\varphi - \pi/4))] \cdot \cos \theta \sin \varphi \end{bmatrix} \quad |\theta| \leq \pi/2, \quad |\varphi| \leq \pi$$

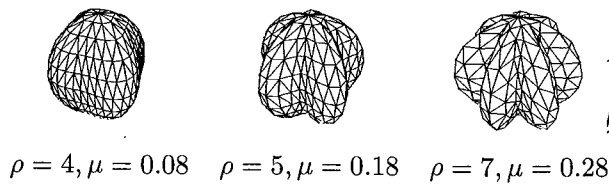


図 3: 対象物体 ( $\nu = 0.2, \tau = D = 1.0$ )

この曲面は  $\theta$ - $\varphi$  表現を変形したもので、 $\nu, \tau$  は縦・横・奥行の比を決めるパラメータ、 $\rho$  は凹凸の数、 $\mu$  は凹凸の深さ、 $D$  は物体の大きさを表し、対象物体については  $D = 1.0$  とする。 $\nu = \tau = \mu = 0$  のときこの曲面は球となる。球面上で点密度がほぼ一定となるように特徴点を設定し、上の式に従って変形する。更に、近接点を結んだ 3 三角形群で曲面を構成し、特徴点の隠蔽判定に用いる。

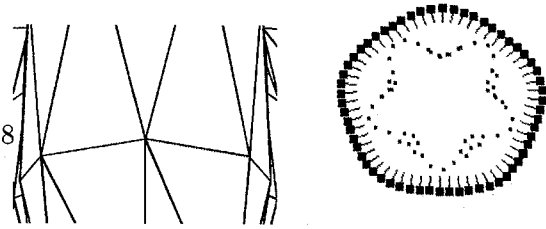
真円からある程度離れたカメラ軌跡を生成するために、視点の決定にも上の曲面の式を用いる。対象物体とは異なるパラメータを用いて、対象物体よりも大きな曲面を作り、 $\theta = 0$  で切った断面の曲線を上下に平行移動させたものに沿って視点を設定する。光軸方向は、この曲線の接線によって決めるものとする。このため、光軸方向は必ずしも物体中心を向いていない。

形状、画像数等を以下のように変化させ、135 通りのサンプルを得た。このうち、画像間で復元に必要な数以上の特徴点对応が得られず理論的に一意な復元が不可能なサンプルを除いたもの（円周撮影 98 通り、疑似広角撮影 115 通り）について、復元の成功/失敗を調べた。

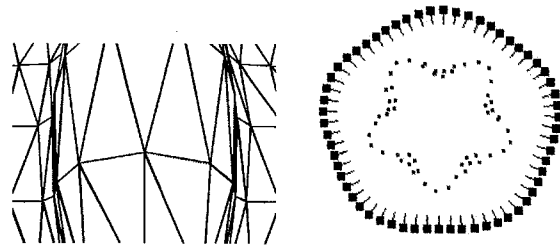
**形状:** 図 3 に示す 4 角形、5 角形、7 角形を用いた。角数の増大と共に物体の凹凸も増加させたため、4 角形の場合ほとんど凹凸はないが、7 角形の場合、物体は深く凹んでおり隠蔽が起りやすい。隠蔽が多いと特徴点对応は少なくなり、復元困難となる。

**特徴点数:** 各形状につき 300, 600, 1200 と変化させた。特徴点数が少ないほど隣接画像での対応数が減り、復元困難となる。

**画像数:** 物体周囲を一回転する 5 角形 ( $\rho = 5, \mu = 0.03, \nu = 0.1$ ) の軌跡に沿って 30, 60, 90ヶ



(a) 最近撮影 Nearest ( $D=1.25$ )



(b) 最遠撮影 Farthest ( $D=1.45$ )

図 4: 5 角形を撮影した例

所から、22.5 度見上げる角度で画像を作成した。各画像の大きさは 600x450pixel で、焦点距離は画像の横幅と同じ（横画角が約 53°）とした。画像数が少ないほど隣接画像間の対応が少なく復元困難となる。

**カメラ-物体間距離:** 距離が遠い場合復元は容易で、カメラが物体に近づくにつれ隣接画像間での対応が減り復元困難となる。ここでは、復元成功率が大きく変化するような 5 段階 ( $D = 1.45, 1.40, 1.35, 1.30, 1.25$ ) の距離を用いた。図 4 に画像および物体とカメラ軌跡の上面図の例を示す。5 段階のうち最も遠い距離の場合でも、各画像には物体の一部しか入らないことがわかる。上面図において周辺部の線付の点はカメラ位置と光軸を表す。その内側の点の集合は物体の概形を表す特徴点である。なお、画像列の 2 画像以上に現れない特徴点は表示されていないため、特徴点は主に物体辺縁部に集中している。実際に復元される特徴点数は、57~427 点となった。

復元には、全回転量が 360 度の注視回転を仮定した初期値を用いた。観測に誤差は加えず、復元結果の再投影誤差 RMS が 0.1pixel 未満の場合を復元成功と判定した。復元結果の特徴点



の3次元座標とシミュレーションにおける真の3次元座標が一致するよう、3次元アフィン変換により復元結果を位置合わせしたところ、成功と判定した復元結果では、平均距離誤差が物体の大きさ ( $D=1.0$ ) に対して1%以下であり、形状が正しく復元できていることを確認した。

図5は円周撮影法において、横軸を平均出現率、縦軸をカメラ-物体間距離としたときの復元成功/修正による成功/失敗の分布を示す。平均出現率が15%以上の全ての場合作で注視回転の初期値から正解が得られた。カメラ-物体間距離が近くなるにつれて、平均出現率も低下してくるが、9%程度でも、解の修正により正解が得られる場合があった。また平均出現率が下がると復元成功率も下がる傾向があり、与えられた問題が解けるかどうかを表す指標として平均出現率が適切であることも確認できる。単純な比較はできないが、Tomasi and Kanadeは、彼らの手法を16%の問題に適用している[7]。平均出現率の観点から本手法は、全周復元の比較的広い範囲の問題に対して、設定した初期値から非線形最適化により正解が得られることが示された。

同様に図6に疑似広角撮影法における復元成功/修正による成功/失敗の分布を示す。画像数は $3 \times 10$ (10ヶ所から各3枚)、 $3 \times 20$ 、 $3 \times 30$ の3通り、他のパラメータは前記実験と同じである。同一位置から複数の画像を撮影しているが、定式化上は、この情報は全く利用しておらず、初期値も円周撮影と同じものを用いた。復元結果の上面図の一例を図7に示す。平均出現率が12%以上の全ての場合作で注視回転の初期値から正解が得られた。また、平均出現率が6%程度でも、正解が得られる場合があった。円周撮影よりも疑似広角撮影の方が、設定した初期値が正解からより離れた問題となっているが、結果的には、平均出現率が低く、カメラ-物体間距離が近い場合でも、円周撮影の場合より成功率が高い。これは、同じ平均出現率でも、疑似広角撮影の方が同一位置からカメラが撮影する範囲の空間分布が広がるために生じる利点と考えられる。

なお、ここでは用いなかったが、疑似広角撮

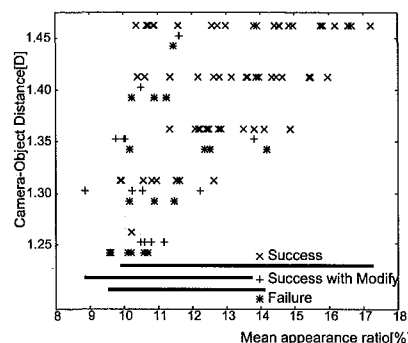


図5: 復元成功/修正による成功/失敗の分布 (円周撮影法)

影では、同一位置から撮影した複数の画像のカメラ位置に対して、カメラ位置が同一であることを利用して定式化することにより、推定すべきパラメータ数が減少し、更に容易かつ高精度に解を求めることが期待できる。このように、全周復元に対しては、円周撮影よりも疑似広角撮影の方が、より適切であると考えられる。

実験においては、解の修正はパラメータ  $L$  の手動設定により行った。解の修正により正解が得られた例のうち、4.3節に示したパラメータ  $L$  の自動設定法のみにより正解が得られるものは、円周撮影の場合14例中2例、疑似広角撮影の場合4例中0例であった。この際、閾値  $T=2.0$  と設定し、前処理付き共役勾配法で前処理行列の再計算を行う16反復毎に、判定を行った。解の修正自体は、局所解から抜け出すためにある程度有効と考えられるが、パラメータ  $L$  の自動設定法については、今後より有効な手法を検討する必要がある。

次に、注視回転による初期値と正解とのずれの大きさが復元の成功/失敗に与える影響を評価するために、撮影において、実際の回転量が360度以下の場合の復元実験を行った。特徴点数600点、カメラ位置60箇所、カメラ-物体間距離が5段階中の3番目 ( $D=1.35$ ) の設定のもとで、復元に用いる画像数を  $F=2, \dots, 60$  枚の間で変化させ、実際の全回転量を  $360 \times F/60$  により設定し、注視回転による初期値 (全回転量360度) から形状復元を行った。初期値と復

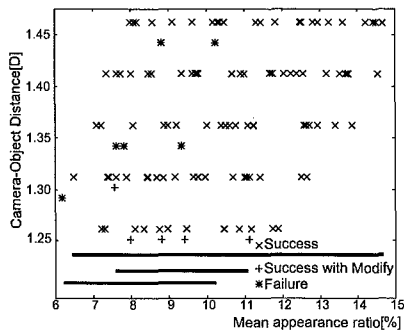


図 6: 復元成功/修正による成功/失敗の分布 (疑似広角撮影法)

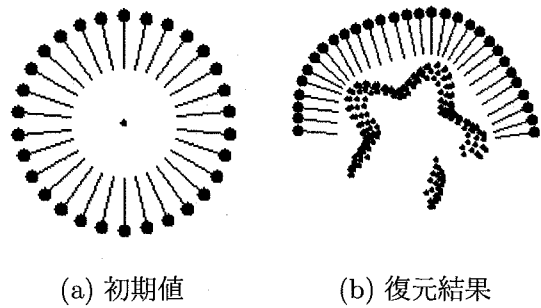


図 8: 回転角 180 度の形状復元

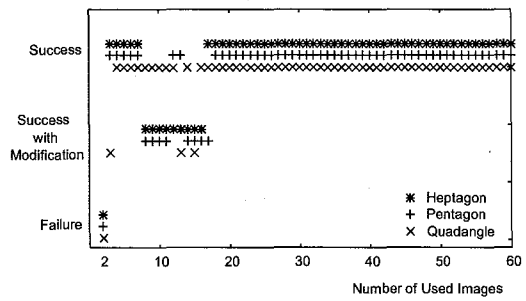


図 9: 実際の回転量と復元の成否の関係

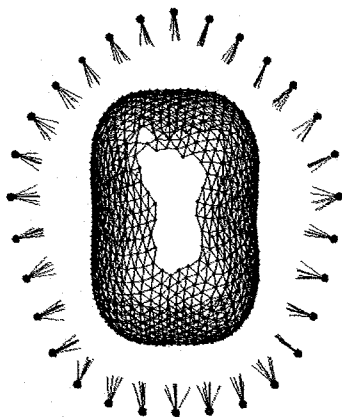


図 7: 疑似広角撮影法による復元結果

元結果の例を図 8 に示す。復元に用いる画像数が 30 枚の場合、初期値 (図 8 左) では、30 台のカメラは円周上に等間隔に並んでいる。一方、復元結果 (図 8 右) では、カメラは円周の半分に位置しており、初期値と正解との間には、円周中心に対して最大 81 度のずれがある。

4 角形、5 角形、7 角形それぞれの場合での、復元成功/修正による成功/失敗の分布を図 9 に示す。横軸は復元に使用した画像数である。いずれの場合でも、画像数 18 枚以上 (初期値との誤差は最大 109 度) では成功している。また、17 枚以下でも、画像数が 3 枚以上 (誤差は最大 114 度) の場合、解の修正により復元が成功している。すなわち、注視回転による初期値は、全周以下の復元問題の初期値としても比較的広い範囲に適用可能といえる。

全周復元問題に対する他の簡便な初期値の例としては、全特徴点が 1 点に縮退しており、かつ、カメラ位置も 1 点に縮退し、特徴点の初期

値方向を向いているというものが考えられる。これは、注視回転による初期値において、カメラ移動の角速度を  $\omega = 0$  と設定した場合に等しい。円周撮影のシミュレーション実験において、復元が成功した 82 例について、この初期値から復元をする実験を行ったが、解の修正を行っても復元が成功する例はなかった。同様に、実際のカメラの移動方向と注視回転による初期値の移動方向を逆に設定した場合も、復元が成功する例はなかった。これらの例では、初期値と正解のカメラ位置の誤差は円周中心に対して最大 180 度近くになるが、このような場合、復元が行えないことが多い。このことから、全周復元問題では、初期値を適切に設定することが重要であり、解の修正法は局所解から抜け出すための補助的な手段と位置付けられることが分かる。

カメラ間隔の偏り、物体位置の偏りなど、他の要因が復元の成否に与える影響については、今後さらに検討する必要がある。

## 5.2 実画像を用いた復元

ここでは、実画像列からの復元結果を 2 例示す。特徴点の抽出および対応付けは、手作業により行った。

図 10 に、原爆ドームを撮影した画像 29 枚中の 2 枚と復元結果の上面図を示す。この実例では、建物から離れた位置から撮影を行っており、平均出現率  $r$  は 17% と高いため、注視回転の初期値から局所解に陥ることなく目視により妥当な復元結果が得られ、解の修正は不要であった。復元結果における再投影誤差は、 $RMS=1.3$  画素未満であった。

図 11 の例では、建物に近い位置から撮影しており、建物のごく一部しか撮影されていない画像を含む。そのため、平均出現率は 6% と低い。また、近距離から広範囲を撮影するため、光軸を上下左右に大きく動かしながら撮影しており、疑似広角撮影となっている。この例では、最適化計算において、再投影誤差が大きな値 ( $RMS=67.8$  画素) のまま減少しなくなり、局所解に陥ったと判断された (図 11 左下)。そ

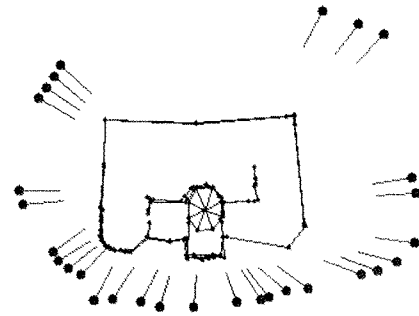


図 10: 原画像の例と復元結果 原爆ドーム ( $P = 122, F = 29, r = 17\%$ )

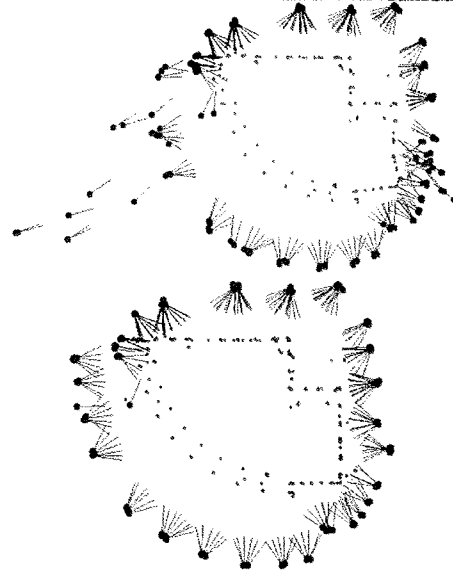


図 11: 原画像の一部と中間結果, 復元結果講堂 ( $P = 189, F = 280, r = 6\%$ )

の後、手動で解の修正を行ったところ、図 11 右下の復元結果が得られた。この時の再投影誤差は  $RMS=5.62$  画素と、多少大きな値であるが、図右上方の建物の垂直な角が良好に復元されているなど、建物全体が良好に復元できていることが確認できる。

その他、100 枚以上からなる全周実画像列 5 例においても良好な復元結果が得られた。

## 6 おわりに

本稿では、建物画像列からの全周形状復元問題を再投影誤差最小化の考えに基づき、非線形最適化により直接解くために、注視回転という簡単なモデルを用いて撮影位置と姿勢の初期値を設定する手法を提案した。さらに、この方法だけでは正解が得られない場合を考慮して、解の修正により局所解から抜け出し正解を得る方法について述べた。

実験の結果、提案した初期値は、その仮定となる注視回転から離れた画像列に対しても、正解を得ることが可能であった。このとき、非線形最適化を適用する際に従来行われていた特徴点位置の初期推定処理は全く必要なかった。また、シミュレーション実験から、全周復元のような隠れ点の多い問題に対しても、全周の 12~15% 程度が撮影された画像からなる画像列から形状復元が行えることが確認できた。このことは、簡便な初期値設定法が、全周復元の広い範囲の問題に適用可能であり、従来、非線形最適化計算の前提としていた複雑な初期値推定は、大幅に簡略化できる可能性があることを示している。

復元においては、円周撮影より、疑似広角撮影の方が、平均出現率が低くても形状復元が成功する傾向があり、全周復元に適していた。これは、後者の撮影法の方が、各特徴点を広範囲から撮影し、拘束が強いためである。本論文では、平均出現率を問題の難しさを表す指標として用いたが、カメラ間の位置関係や個々の画像毎の特徴点の出現率の影響など他の指標も導入することにより、形状復元が正しく行える条件を明確にし、より良い撮影方法を検討していく

必要がある。また、より大規模な実問題に本手法を適用すること、および、特徴点の抽出と対応付けを自動化することも今後の課題である。

## 参考文献

- [1] B. Triggs, P.F. McLauchlan, R.I.Hartley, A. W. Fitzgibbon: "Bundle Adjustment – A Modern Synthesis," *Vision Algorithms: Theory and Practice*, LNCS 1883, Springer, pp.298–375 (2000).
- [2] R. Szeliski and S. B. Kang: "Recovering 3D Shape and Motion from Image Streams using Non-Linear Least Squares," *CVPR*, pp.752–753, (1993).
- [3] M. Pollefeys, F. Verbiest, and L. V. Gool: "Surviving Dominant Planes in Uncalibrated Structure and Motion Recovery," 7th *ECCV*, pp. II-837–851, (2002).
- [4] 右田剛史, 天野晃, 浅田尚紀: "非線形最適化計算を用いた運動からの形状復元," 画像の認識・理解シンポジウム (MIRU2000) 論文集, pp. II-199–204, (2000).
- [5] 右田剛史, 天野晃, 浅田尚紀: "3次元形状・運動復元のための高速非線形最適化計算法," *情報処理学会論文誌*, vol.44, no.11, pp.2864–2872, (2003).
- [6] P. F. McLauchlan: "A Batch/Recursive Algorithm for 3D Scene Reconstruction," *CVPR*, pp.738–743, (2000).
- [7] C. Tomasi, T. Kanade: "Shape and Motion from Image Streams under Orthography: a Factorization Method," *IJCV* 9(2), pp. 137–154, (1992).
- [8] P. Strum, and B. Triggs: "A Factorization Based Algorithm for Multi-Frame Projective Structure and Motion," *ECCV*, pp. 709–720, (1996).
- [9] 植芝 俊夫, 富田 文明: "奥行きパラメータの逐次推定による多視点透視投影画像のための因子分解法," *信学論*, vol.J81-D-II, no.8, pp. 1718–1726, (1998).
- [10] 藤木 淳, 蔵田 武志: "計量アフィン射影モデルの逐次型因子分解法" *信学論*, vol.J84-D-II, no.8, pp. 1663–1673, (2001).

- [11] D. Jacobs: "Linear fitting with missing data: Application to structure-from-motion and to characterizing intensity images," CVPR, pp. 206–212, (1997).
- [12] H. Aanas, R. Fisker, K. Aström and J. Carstensen: "Robust Factorization," PAMI 24-9, pp. 1215–1225, (2002).
- [13] H. C. Longuet-Higgins: "A computer algorithm for reconstructing a scene from two projections," Nature, 293:133-135, (1981).
- [14] R. I. Hartley: "Lines and Points in Three Views and Trifocal Tensor," IJCV 22(2), pp. 125–140, (1997).
- [15] 佐藤 智和, 神原 誠之, 横矢 直和, 竹村 治雄: "マルチベースラインステレオ法を利用した動画像からの屋外環境の三次元モデル化," バーチャルリアリティ学会論文誌, Vol.7, No.2, pp.275–282, (2002).
- [16] R. Szeliski and S. B. Kang: "Shape Ambiguities in Structure from Motion," PAMI 19-5, pp. 506–512, (1997).
- [17] J. Weng, N. Ahuja and T. S. Huang: "Optimal motion and structure estimation," PAMI 15-9, pp. 864–884, (1993).

# Large Object Shape Recovery from Uncalibrated Camera Motion by Non-Linear Optimization

## Abstract

3D recovery of object shape and camera motion from 2D image sequence is formulated as a non-linear optimization problem. Such a non-linear approach indeed ensures the most precise solution, but it involves computational difficulties, i.e. complexity and instability.

This paper describes a practical procedure to solve the non-linear optimization problem for object shape and camera motion recovery. First, we discuss the imaging model for large objects such as buildings; one shot and pseudo wide angle models, where the camera motion is modeled as a uniform motion along circular path with gazing at the center. Then, the generalized initial values for circular motion is proposed and “mean appearance rate (MAR)” of feature points is defined as a measure of stability.

Experimental results with synthetic images show that 3D recovery was stably performed from image sequences with MAR over 15%, quasi-stable with over 8.5% for one shot model, and stable with over 10.3%, quasi-stable with over 4.4% for pseudo wide angle model. 3D shape and motion using real images were recovered successfully from MAR 17% for one shot model and 6% for pseudo wide angle model. These results have demonstrated the robustness of the generalized initial values and effectiveness of MAR for large object shape from uncalibrated camera motion problem.

## Key Words

3D shape recovery, non-linear optimization, uncalibrated camera, circular motion.

## 1 Introduction

3D modeling of real objects is the central concern in computer vision, and many works have been done so far to recover 3D information from image sequences. 3D recovery of object shape and camera motion is formulated as a non-linear optimization problem [1]. Such a non-linear approach indeed ensures the most precise solution, but it involves computational difficulties, i.e. complexity and instability. The iterative search in solution optimizing process often falls into a local minimum unless appropriate initial values are given. Thus, the non-linear method is used as a bundle adjustment [2, 3] of a “good” approximation given by linearized methods [4, 5, 6].

This paper describes a practical procedure to solve the non-linear optimization problem for object shape and camera motion recovery. The target of our study is large objects such as buildings, thus the image sequence is generally taken from a camera that moves around the object. This camera motion is modeled into two types; one shot and pseudo wide angle imaging models as shown in Fig. 1. These camera motions, especially close to the object, necessarily pose the occlusion problem; that is, the object is partially

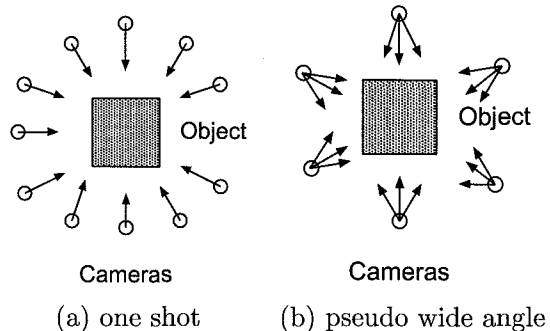


Fig.1 Imaging model for circular motion.

observed in each image. This kind of image sequence includes poor information for 3D recovery, thus the non-linear computation becomes seriously unstable.

In this paper, we propose simple initial values which is robust for shape recovery from circular motion image sequence, and define “mean appearance rate” of feature points as a measure of stability.

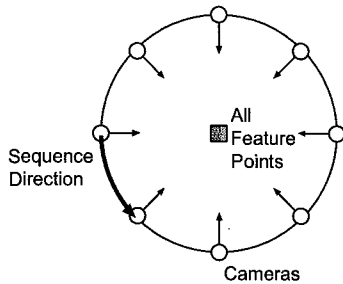


Fig.2 Generalized initial values for circular motion.

## 2 3D Recovery by Non-Linear Optimization

Suppose objects are represented by a set of feature points whose 3D coordinates  $\mathbf{s}_p$  ( $p = 1 \dots P$ ) is defined in the world coordinates, and those feature points are projected onto the  $f$ th image ( $f = 1 \dots F$ ). 2D coordinates of the projected  $p$ th point on the  $f$ th image is denoted by  $\mathbf{u}_{fp}$ . Note that when the  $p$ th point is not observed from the  $f$ th camera, the corresponding  $\mathbf{u}_{fp}$  is undefined in the  $f$ th image.

The pose and position of the  $f$ th camera are represented by the rotation matrix  $R_f$  and translation vector  $\mathbf{t}_f$ , and the perspective projection operator  $\mathcal{P}$  is defined by  $\mathcal{P}(x, y, z)^T = \frac{l}{z}(x, y)^T$ , where  $l$  is the focal length of the camera. Using these notations, 3D recovery of object shape and camera motion is formulated by minimizing the error function  $E(\mathbf{x})$ , as follows.

$$E(\mathbf{x}) = \sum_{(f,p)} |\mathcal{P}[R_f \mathbf{s}_p + \mathbf{t}_f] - \mathbf{u}_{fp}|^2 \quad (1)$$

where  $\mathbf{x}$  denotes a composite vector consisting of  $\mathbf{s}_p$ ,  $\mathbf{q}_f$ ,  $\mathbf{t}_f$ , where  $\mathbf{q}_f$  is a quaternion representation of the rotation matrix. Since Eq. (1) includes non-linear representation such as perspective projection, the problem is solved by using non-linear optimization method, e.g. Levenberg-Marquardt algorithm [7], preconditioned conjugate gradient one [8] and so on. Note that this paper focuses on the stability of solving the non-linear problem, thus we assume for simplicity that the feature point correspondence between images is given properly.

In order to solve the non-linear optimization problem stably, we need appropriate initial values that lead to the global minimum of error function  $E(\mathbf{x})$ . Large objects such as buildings are the target of our study, and thus the image sequence is generally taken from a camera that

moves around the object. This camera motion is modeled as a uniform motion along circular path with gazing at the center where the object is located. Based on this idea, we define the generalized initial values for circular motion, as follows.

$$\begin{cases} R_f = R(\omega f) \\ \mathbf{t}_f = (0, 0, r)^T \\ \mathbf{s}_p = (0, 0, 0) \end{cases} \quad (2)$$

where  $R_f$  and  $\mathbf{t}_f$  denote the pose and position of the  $f$ th camera which is located on a circle of radius  $r$  with gazing at the center, and  $\omega$  indicates the motion direction and step of every one frame, as illustrated in Fig. 2. Since the object shape is unknown at the beginning, the initial shape is given by  $\mathbf{s}_p = (0, 0, 0)$  which means all feature points meet at the center of the circle; that is, object is reduced to a point. While this setting of initial values is very simple, it works successfully for many image sequences taken by the circular motion, as mentioned below.

Here, we introduce a measure of stability called “mean appearance rate (MAR)” that is defined by the mean ratio of each feature point observed throughout the image sequence. This measure is similar to the fill-ratio of measurement matrix used in factorization [6]. When the camera is closer to the object, the smaller part is observed with more occlusions, thus MAR decreases. In such a situation, since the feature points include poor 3D information, shape and motion recovery becomes unstable.

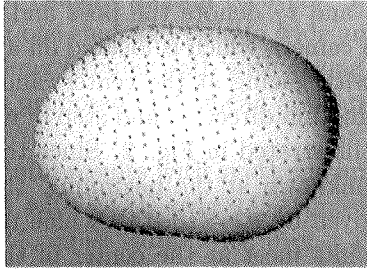
## 3 Experiments

We have performed experiments with synthetic and real image data. In the simulation study, object shape is defined by the following curved surface whose complexity is determined by  $\nu$ ,  $\tau$ ,  $\rho$  and  $\mu$ .

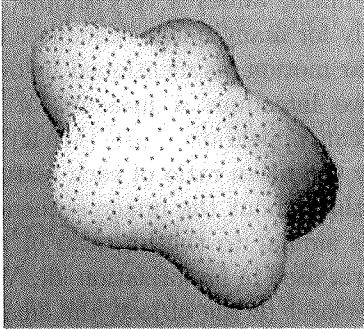
$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} [1 - \nu + \mu \cos(\rho(\varphi - \pi/4))] \cdot \cos \theta \cos \varphi \\ [1 - \tau] \cdot \sin \theta \\ [1 - \nu + \mu \cos(\rho(\varphi - \pi/4))] \cdot \cos \theta \sin \varphi \end{bmatrix}$$

where  $|\theta| \leq \pi/2$ ,  $|\varphi| \leq \pi$ , and the  $x$ - $z$  plane is assumed as the ground.  $\nu$  and  $\tau$  define the ratio of height to width,  $\rho$  and  $\mu$  denote the number and depth of the concavity, respectively, and thus the object becomes a sphere when  $\nu = \tau = \mu = 0$ .

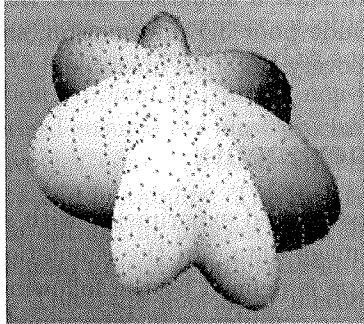
Figure 3 shows three examples of rectangular, pentagonal, and heptagonal star shapes with



$\rho = 4, \mu = 0.08$

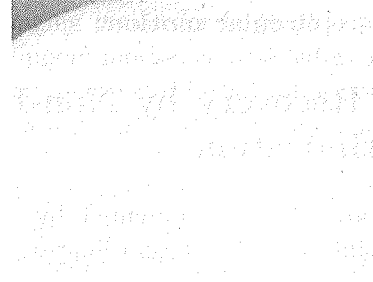
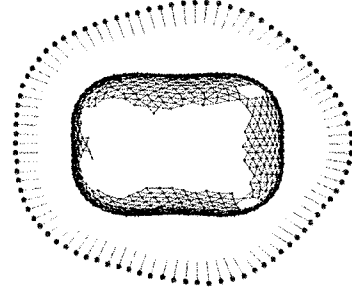


$\rho = 5, \mu = 0.18$



$\rho = 7, \mu = 0.28$

**Fig.3** Objects for simulation study.



**Fig.4** Camera path around the rectangular object and an image nearest to the object.

black feature points on the surface. We generated 135 kinds of image sequences from the combination of 3 object shapes, 3 numbers of feature points, 3 numbers of frames, 5 distances between object and camera, as follows.

**shape** : three kinds of shapes shown in Fig. 3 are used. Note the deeper concavity makes the more occlusions.

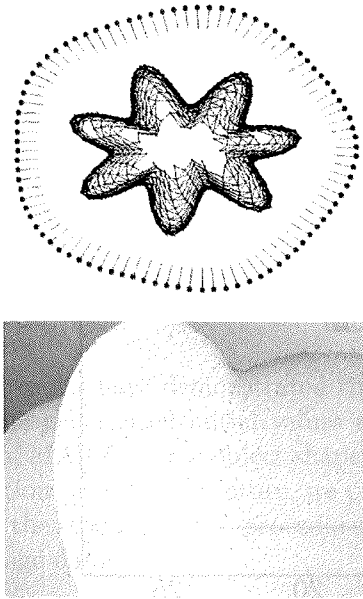
**feature points** : 300, 600, and 1200 points are spread over the surface. Note the fewer points represent the less 3D information.

**frames** : 30, 60, and 90 images consisting of  $600 \times 450$  pixels are taken on the pentagonal path around the object with 22.5 degree angle of elevation. Note the fewer frames include the less corresponding points between images.

**distance** : 5 different radii are used to draw the camera path. Figure 4, 5 show examples of the nearest and farthest images of rectangular and heptagonal objects. Note the whole shape is not observed even in the farthest case shown in Fig. 5.

The stability of each image sequence is evaluated by using Eq. (1) with the initial values for circular motion, and we classified the results into three categories;





**Fig.5** Camera path around the heptagonal object and an image farthest to the object.

**stable** : the correct solution is obtained by automatic convergence, where the criterion of success is that RMS error of Eq. (1) become less than 0.1 pixel.

**quasi-stable** : the correct solution is obtained by retrieval, where the first recovery is performed by using the higher MAR subset of feature points, then the result is used as initial values for the next recovery using all feature points.

**unstable** : there is no way to obtain correct solution from the initial values for circular motion.

Figure 6, 7 show the results of 3D recovery using one shot and pseudo wide angle imaging models, respectively. The abscissa denotes the mean appearance rate (MAR), and the ordinate shows 5 distances between object and camera.

From Fig 6, we found that the image sequence whose MAR is over 15% yield correct shape stably, and that the lower MAR makes the lower success rate. This suggests that the MAR is available as a measure of the stability. In the pseudo wide angle imaging case, Figure 7 shows almost the same distribution of stable, quasi-stable, and unstable but the stability is improved in comparison with the one shot imaging case. The MAR over 10.3% becomes stable, and

we can obtain the correct solution from MAR at least 4.5%.

Experiments with real images were performed by using seven image sequences, two of which are shown in Fig. 8, 9. Figure 8, the Hiroshima atomic bomb dome, includes two images in 29 frames and the top view of recovered shape with camera motion. In this case, images were taken from outside of the security fence, and MAR was 17% which is high enough to recover the 3D information stably even using the one shot imaging model. Figure 9, the university auditorium, shows two image in 280 frames and the recovered shape with camera motion. In this case, images were taken from close positions, and thus MAR was 6% which is quite low for stable recovery. However, we have the correct solution by using the pseudo wide angle imaging model. All of the other five image sequences consisting of over 100 frames yield correct solutions successfully from the generalized initial values for circular motion.

## 4 Conclusion

This paper described a practical procedure to solve the non-linear optimization problem for recovering large object shape from image sequence taken by a camera moving around the object. We proposed the generalized initial values for circular motion and defined mean appearance rate (MAR) of feature points as a measure of stability.

Using the proposed initial values, we have performed experiments with synthetic and real image data, and have had the following findings;

- In the simulation study, 3D recovery was stably performed from image sequences with MAR over 15%, quasi-stable with over 8.5% for one shot model, and stable with over 10.3%, quasi-stable with over 4.4% for pseudo wide angle model.
- In the real image case, 3D shape and motion were recovered successfully from MAR 17% for one shot model and 6% for pseudo wide angle model.
- In most objects even with complicated shape, 3D shapes are recovered successfully from the simple initial shape reduced to a point.

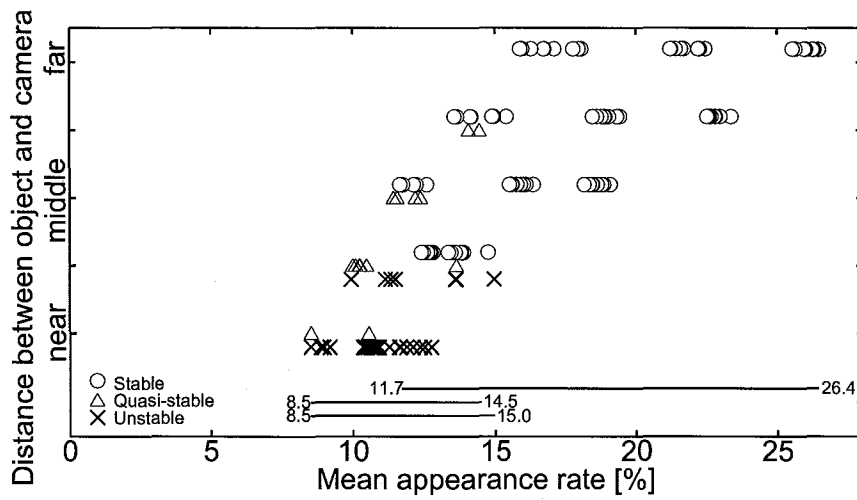


Fig.6 Result of shape recovery using one shot imaging model. (○:stable, △:quasi-stable, ×:unstable)

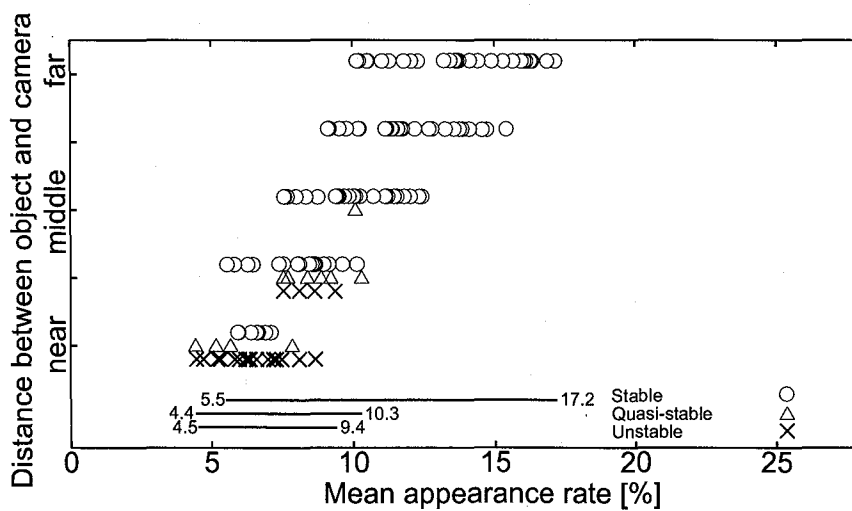


Fig.7 Result of shape recovery using pseudo wide angle imaging model. (○:stable, △:quasi-stable, ×:unstable)

- 3D information is recovered from inaccurate total angle of rotation as initial setting, but the counter rotation tends to failure. For example, 270 degree, the true angle of rotation, is converged from 360 degree as initial angle.
- The pseudo wide angle imaging model yields stabler results than the one shot model.

These results have demonstrated the robustness of the generalized initial values and effectiveness of MAR for large object shape from circular motion problem. In future, we plan to investigate the stable recovery procedure from low MAR image data.

## References

- [1] R. Szeliski and S. B. Kang: Recovering 3D Shape and Motion from Image Streams using Non-Linear Least Squares, *CVPR*, 752–753, 1993.
- [2] B. Triggs, P.F. McLauchlan, R.I.Hartley, A. W. Fitzgibbon: Bundle Adjustment – A Modern Synthesis, *Vision Algorithms: Theory and Practice*, LNCS 1883, Springer, 298–375, 2000.
- [3] M. Pollefeys, F. Verbiest, and L. V. Gool: Surviving Dominant Planes in Uncalibrated Structure and Motion Recovery, *Proc. ECCV*, (II), 837–851, 2002.
- [4] H. C. Longuet-Higgins: A computer algorithm for reconstructing a scene from two projections, *Nature*, 293, 133-135, 1981.
- [5] R. I. Hartley: Lines and Points in Three Views and Trifocal Tensor, *IJCV*, 22(2), 125–140, 1997.
- [6] C. Tomasi, T. Kanade: Shape and Motion from Image Streams under Orthography: a Factorization Method, *IJCV*, 9(2), 137–154, 1992.
- [7] D.W. Marquardt,: An Algorithm for Least-squares Estimation of Nonlinear Parameters, *SIAM J. Appl. Math.*, 11, 431–441, 1963.
- [8] A. Amano, T. Migita, N. Asada: Stable Recovery of Shape and Motion from Partially Tracked Feature Points with Fast Nonlinear Optimization, *Proc. Vision Interface*, 244–251, 2002.

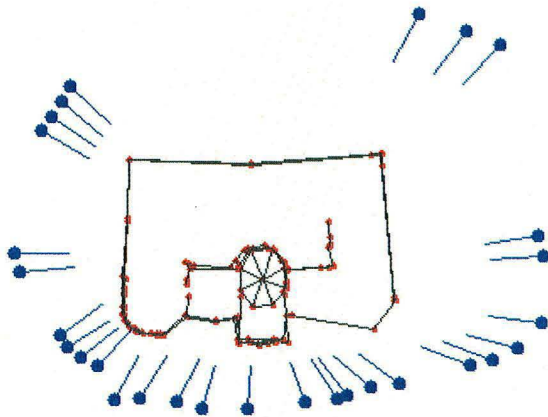
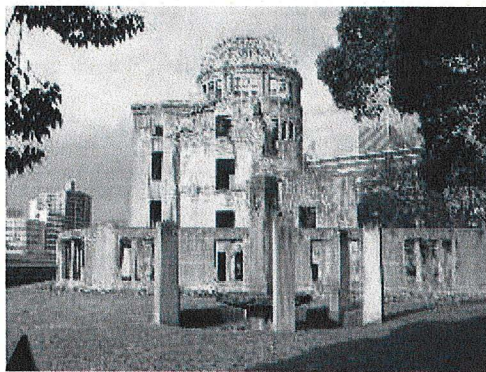


Fig.8 Hiroshima atomic bomb dome: two images in 29 frames and recovered shape with camera motion (top view).

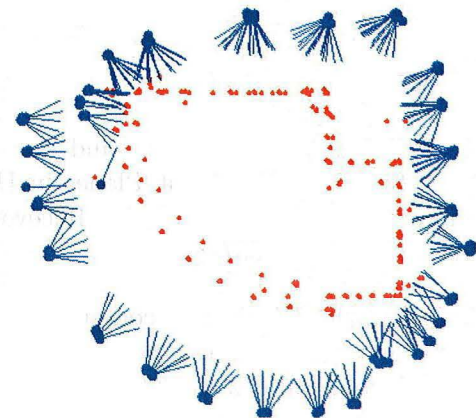
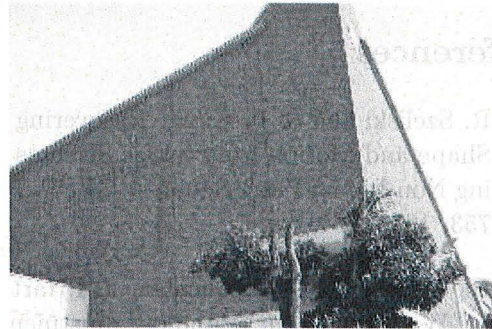


Fig.9 University auditorium: two images in 280 frames and recovered shape with camera motion (top view).

# モザイクングを用いた疑似広角撮影画像列からの形状復元の評価

## 1 はじめに

近年、インターネットの普及に伴い、デジタルアーカイブとして保存された芸術作品や文化財のウェブ上での公開が注目されている [1]. 文化財をデジタルアーカイブで公開するためには、その3次元モデルを生成する必要がある. しかし、特殊な計測機器を用いて3次元モデルを生成する手法は、生成コストや対象物体の大きさが制限されるなどの問題がある. 一方、複数視点から撮影された画像列から3次元モデルを生成する手法では、特殊な機器が不要であり、対象物体の大きさが制限されることもない. そのため、画像列からの3次元形状復元 (Structure from Motion) の研究が盛んに行われている [11, 12, 13, 14, 15, 16].

画像列からの形状復元では、視点の異なる画像間で対応する点を対応付ける必要がある. 対応点を正確に与えることができれば、透視投影のモデルに基づき各点の3次元座標が計算可能である. 特に、特徴点が多数の画像上で対応付けされていれば、特徴点の3次元座標とカメラ位置・姿勢の信頼性は高くなる. このため、対象物体全体が画像上に撮影されていることが望ましいが、建造物のような大きな物体が対象の場合は撮影時に対象物体と十分な距離をとれないことが多く、これは困難となる. この問題に対し、各カメラ位置において、カメラ方向を変化させて複数の撮影を行う疑似広角撮影により得た画像列を用いて、復元する手法がある [3]. 疑似広角撮影された画像列では、各カメラ位置において、より広範囲の情報を得ることができるが、各画像で対象物体の一部分しか撮影されていないため画像間での特徴点の対応付けが行いにくいという問題が生じる. 加えて、円周撮影を行った場合と比べて、画像枚数が増加するため、求めるカメラパラメータの量が増加してしまい、一度に形状復元結果を得ることが難しい.

そこで本研究では、疑似広角撮影画像に対してモザイクングを用いることで、形状復元を行う手法を提案する. モザイクングは、複数の画像を接合し、視野が広く解像度の高いパノラマ画像を合成する技

術であり、オプティカルフロー (画像間の見かけの速度場) を検出して輝度値の差を最小化する方法が多数提案されている [17, 18, 19]. ただし、カメラ運動が比較的自由的な画像間や、輝度の異なる画像間では、単純なオプティカルフロー検出では対処できない. これを解決するために、画像間で特徴点の検出・対応付けを行うことで、画像間を適切に合わせる平面射影変換パラメータを推定する手法が提案されている [4, 7, 8, 20]. ただし、これらの手法では、画像情報のみからエピポラ方程式を求めているため、物体の見え方の変化や隠蔽、フレームアウト、繰り返しパターンなどにより特徴点の対応誤りを生じる可能性がある. 特徴点の対応付けに誤対応が含まれると正しく平面射影変換パラメータが推定されない. このため、本手法では、大局的なマッチングによる特徴点の対応付けを用い、パノラマ画像を生成する [2]. この手法は、2画像間の変形をアフィン行列で表せると仮定し、特徴点の対応付けにおける大局的な幾何制約として用いる. これにより対応候補の探索範囲を限定し、誤対応問題を軽減している. また、最適な平面射影変換パラメータを求める際に、単純に対応点の組合せを全探索しては処理時間が長くなってしまふ. そこで、最適な平面射影変換パラメータを得られる対応点探索を高速化する手法を提案する.

次に、生成したパノラマ画像列を利用して形状復元を行う. 従来、モザイクングを用いた形状復元は、評価実験されておらず、実際に形状復元に対し有効なのか解明されていない. そこで評価実験を行い、モザイクングを用いた形状復元の有効性についての検証を行う.

以下、2節では透視投影や疑似広角撮影などいくつかの諸定義についてまとめる. 続いて3節では、大局的な制約を用いたモザイクング手法と形状復元方法について述べる. また、4節では本手法をシミュレーション画像と実画像に適用し有効性を評価する. 最後に5節では、本論文のまとめを行う.

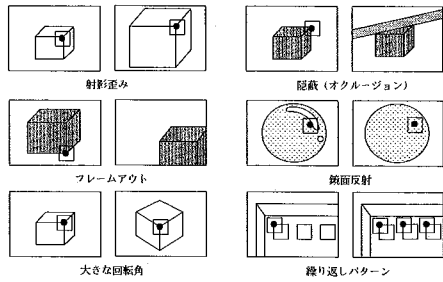


図 1: 特徴点類似度における一般的な問題点

## 2 疑似広角撮影画像列からの形状復元

### 2.1 特徴点類似度における一般的な問題

カメラ運動から形状復元を行う場合に問題となる点として、以下のようなものが挙げられる。

#### 1. 射影歪み

同じ大きさ、同じ向き of 物体であっても、カメラからの距離によって異なる大きさに投影されることを指す。

#### 2. 隠蔽 (オクルージョン)

隠蔽とは、カメラに対して撮影対象の一部が撮影対象の裏側、もしくは別の対象物の後方に隠れて見えないことを指す。

#### 3. フレームアウト

建物や物体を近い位置から撮影したとき、カメラ前方にあるにも関わらず画像内に収まらない撮影対象の部分が生じることをさす。

#### 4. 鏡面反射

光が拡散せず、物体への入射角と同じ角度で反射することを指す。鏡面反射領域では 2 画像間で対応する領域に大きく異なる輝度値が存在する。

#### 5. 大きな回転角

画像間で物体に大きな回転が含まれていることを指す。物体を視点を変えて見ると、特徴が変形して見える。

#### 6. 繰り返しパターン

画像中に似た形状のパターンが多く存在する

場合を指す。局所的には特徴の見分けがつかない。

これらは実画像を用いた場合、頻繁に起こりうる問題である。

これらの問題点のうち、フレームアウトを解消する手段として、撮影時に、1 視点から向きを変えて複数枚の画像を撮影し、フレームアウトにより撮影されなかった部分を撮影する方法がある。その方法について、次節で説明する。

### 2.2 疑似広角撮影

対象物体全体の形状を復元するには、対象物体の全周囲を撮影した画像列が必要である。この場合、単純には図 2(a) のように対象物体の周囲を円状移動しながら、円の中心方向にある対象物体を撮影する方法が考えられる。この撮影モデルを円周撮影とよぶ。円周撮影では、対象物体の全周囲の画像列が得られるが、現実の建造物等対象物体である場合、対象物体とカメラの間の距離を十分に離して撮影できないことが多い。その場合、円周撮影した画像 1 枚では、対象物体がフレームアウトしてしまい、全体の形状復元が行えない。

この問題を解決するために、各カメラ位置において、カメラ方向を変化させて複数の撮影を行う疑似広角撮影 (図 2(b)) により画像列を得る。これにより、各カメラ位置において、より広範囲の情報を得ることができる [3]。各カメラ位置で、対象物体全体の情報が得られるよう、疑似広角撮影を行うことにより、対象物体全体の形状復元が行える。

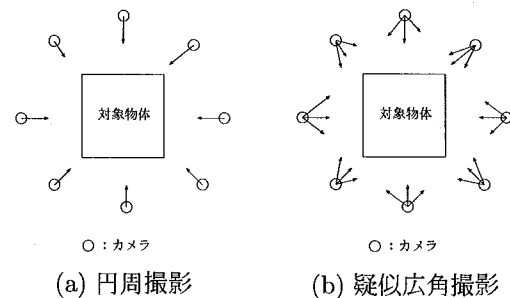


図 2: 撮影方法



## 2.3 疑似広角撮影画像からの形状復元における問題点

疑似広角撮影では、円周撮影に比べ、同一の視点数でも得られる情報が多くなるため、フレームアウトの問題に対処でき、形状復元も安定して行える。

一方で、1画像に撮影されている対象物体の量は少なくなっている。そのため、画像間において特徴点の対応付けを行う際に、利用できる情報量が少なく、対応付けが困難となる。また疑似広角撮影では1視点で複数枚撮影を行っているため、求めるべきカメラ位置の数が増加する。一般に、パラメータ推定では、求めるべきパラメータの数が増えるにつれて、推定が困難となる。そのため、疑似広角撮影した画像を個別の画像として扱い、円周撮影と同じ方法で形状復元する処理では、形状復元が不安定になるという問題がある。

このような問題を解決するために、同一視点から撮影された画像に対しモザイクングを行い、1枚のパノラマ画像を作成する。そして、作成されたパノラマ画像列を用いて形状復元を行う。同一視点で撮影された画像を1枚のパノラマ画像にすることにより、1枚の画像の中に対象物体の全対象が撮影されているため、画像間での対応付けがとりにくいという問題が解消される。また、撮影画像は1視点につき1枚となるため、求めるべきカメラパラメータの数も円周撮影を行った場合と同じになる。そのため、求めるべきカメラパラメータ数の増加という問題も解決できる。

以上を踏まえて、上記の問題を解決するために、本研究では疑似広角撮影で得られた画像に対しモザイクングを行いパノラマ画像を作成し、その画像から形状復元を行う手法を提案する。

## 3 形状復元におけるモザイクングの利用

### 3.1 モザイクング

モザイクングとは、複数枚の画像列を重ね合わせて、視野の広い1枚のパノラマ画像を合成する技術であり、古くから航空写真や衛星写真、リモートセンシング画像の合成などに利用されている。近年では、臨場感の高い仮想環境を構築する手法として

の利用や、監視システム、映像ブラウジングインタフェースといった分野などにも利用されている。

モザイクングにおける画像を接合する原理は、同一視点で向きのみ変化させて撮影した画像間、または異なる2視点から撮影した平面の画像間が「射影変換」により結ばれるという事実である。また、平面ではないシーンを撮影した画像も、十分遠方であれば平面シーンを撮影したとみなすことができる。したがって、一方の画像に射影変換を施すことにより、他方の画像と重ね合わせることが可能である。

平面射影変換を使用した2画像のモザイクングは、主に以下のような手順により行われる。

1. 基準画像と別視点画像において特徴点を検出・対応付ける
2. 対応点を用いて画像間の平面射影変換行列を算出する
3. 平面射影変換行列を用いて、別視点画像を基準画像に重ね合わせる

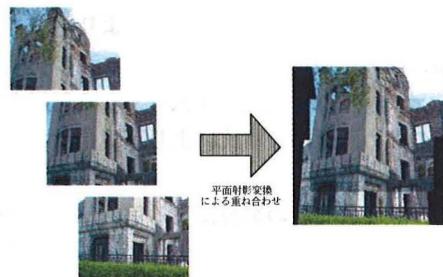


図 3: モザイクング

次節から、各処理の詳細について述べる。

### 3.2 画像間の大局的幾何制約を用いた特徴点の対応付け

本研究では、数十枚の画像列を自動処理するため、明らかな誤対応を確実に排除する必要がある。そのため、まず画像全体を幾何学的変換により直接対応させ、対応付けられた画像の近傍領域に対応する特徴点が存在するか判定する。画像そのものを対応させる幾何学変換は、過度の自由度があると計算が不安定となる欠点がある。そこで、本手法では自

由度6のアフィン変換を、2画像間の変形を表す変換とする。

### 3.2.1 アフィン変換行列の推定

アフィン変換は、平行移動、回転、拡大縮小、スキューの組み合わせとして表現できる画像の幾何学的変換である。アフィン変換行列  $A$  を用いると、画像  $I$  上の座標  $(u, v)$  と画像  $I'$  上の座標  $(u', v')$  の対応は、次式で表される。

$$\begin{aligned} [u' \ v' \ 1]^T &= A [u \ v \ 1]^T, & (1) \\ \text{ここで } A &= \begin{bmatrix} x_0 & x_1 & x_2 \\ x_3 & x_4 & x_5 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

このため、2画像間のアフィン変換行列  $A$  は、対応する画素値の差の2乗和  $E$  の最小化を行う次式で求められる。

$$\arg \min_A E, \text{ここで } E = \frac{\sum_{(u,v)} |I'(u',v') - I(u,v)|^2}{n} \quad (2)$$

ここで、 $n$  は2画像の重なっている画素数である。式(2)により差の2乗和を最小にする  $(x_0, x_1, x_2, x_3, x_4, x_5)$  を求めればよい。これは、勾配法や準Newton法のような解の反復改良手法により解くことができる。なお、並進を表す変数  $(x_2, x_5)$  の値は、変数  $(x_0, x_1, x_3, x_4)$  と比較して大きくなる可能性が高い。このため画像の縦横比により正規化しておく必要がある。

なお、ノイズなどの影響による局所解を避け低周波成分によるマッチングを行うため、あらかじめ平滑化を行った画像を利用する。

### 3.2.2 並進成分の推定

2画像を対応付けるアフィン変換行列の推定は、一度に推定するパラメータが多いため局所解に陥りやすい。特に並進成分に関しては画像サイズに依存するため、回転成分やスケール成分とともに並進成分を推定することは困難となる。このため、まず、あらかじめ全域探索を行うことで2画像間の並進成分を求める。次に求めた並進成分を初期値としてアフィン変換行列を推定する。並進成分の推定方法は次のように行う。

2画像間の変形を並進であると仮定し、並進変換行列  $T = \begin{bmatrix} 1 & 0 & t_u \\ 0 & 1 & t_v \end{bmatrix}$  を用いると、画像  $I$  上の

座標  $(u, v)$  と画像  $I'$  上の座標  $(u', v')$  の対応は、次式で表せる。

$$[u' \ v' \ 1]^T = T [u \ v \ 1]^T \quad (3)$$

2画像間の並進変換行列  $T$  は、対応する輝度値の差の2乗和が最小となるように次式によって求める。

$$\arg \min_T \frac{\sum_{(u,v)} |I(u,v) - I'(u',v')|^2}{n} \quad (4)$$

ここで、 $n$  は2画像の重なっている画素数である。実際には、並進成分  $[t_u, t_v]^T$  を1画素単位で変化させながら、式(4)を計算する全域探索により  $T$  を求める。ここで、探索範囲は、2画像の重なりが画像の大きさの1/4以上である範囲とする。

### 3.2.3 並進変換とアフィン変換を用いた特徴点の対応付け

画像間の全域制約として並進変換とアフィン変換を用いた特徴点の対応付けは、次のようになる。

1. 3.2.2節の全域探索により画素毎に並進成分のみによる画像間対応付けの評価値を計算する。
2. 得られた並進成分を初期値として、3.2.1節の手法を用いてアフィン変換行列を推定する。
3. SSDを用いて画像  $I$  上の特徴点を、画像  $I'$  上の特徴点から特徴点類似度の高い順に1対  $m$  対応<sup>1</sup> させる。
4. 図4に示すように、求めたアフィン変換行列  $A$  を用いて画像  $I$  上の各特徴点の座標変換を行う。座標変換された領域の近傍内に、対応付けている特徴点が存在していれば、特徴点の対応付けを行う。ここで、近傍内に2点以上存在する場合は、より特徴点類似度の高い特徴点を選択する。

以上より、特徴点の対応付け結果を得る。

## 3.3 平面射影変換パラメータの算出

平面射影変換は、以下の式(5)を用いて、8パラメータにより表すことができる。

<sup>1</sup> 実験では、1対5対応とした



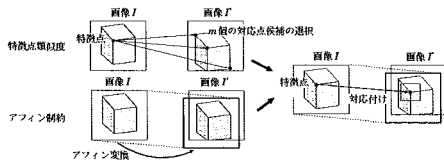


図 4: アフィン制約を用いた特徴点の対応付け

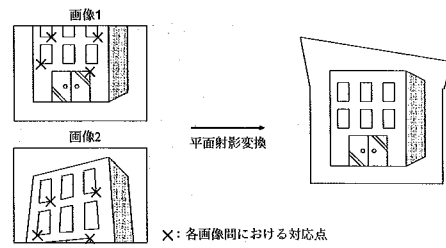


図 5: 平面射影変換によるモザイク

$$\begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix} \sim \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{pmatrix} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \quad (5)$$

ここで、式 (5) は、式 (6) に変換でき、対応点が 4 組以上あれば連立方程式により平面射影変換を求めることができる。

$$\begin{cases} u' = \frac{au+bv+c}{gu+hv+1} \\ v' = \frac{du+ev+f}{gu+hv+1} \end{cases} \quad (6)$$

対応点が 4 組以上ある場合は、以下のような目的関数式 (7) が、最小となる対応点の組を最適な対応点の組として算出する。ここで、基準画像  $I$  上の座標  $(u, v)$  の点の輝度値を  $I(u, v)$ 、平面射影変換  $H_{i,j,k,l}$  により  $P$  へ変換される画像  $I'$  上の座標  $(u', v')$  の点の輝度値を  $I'(u', v')$  とする。

$$E = \sum_{u,v} |I(u, v) - I'(u', v')|^2 \quad (7)$$

### 3.3.1 平面射影変換

アフィン変換は、射影変換の特別な場合である。アフィン変換で行える変形である平行移動、回転、拡大・縮小、せん断に加え、平面扇型変形変換でも行える。

基準画像上の特徴点を  $P(x_1, y_1)$ 、重ね合わせる画像上の  $P$  に対応する特徴点を  $Q(x_2, y_2)$  とする。このとき、平面射影変換で重ね合わされるこの 2 点の関係は式 (5) のように 8 パラメータにより表すことができる。

### 3.3.2 対応点の領域分割を用いた高速化

最適な対応点の組を求めるためには、対応点の 4 組の組合せすべてについて、式 (7) を計算する必要がある。対応点の組の数を  $N$  とするとこの組合せは、 $O(N^4)$  であり、 $N$  が増えると処理時間が長くなる。

一方、平面射影変換を求める際、重なり部分においてできるだけ分散している 4 点の対応点を用いると、最適である平面射影変換が得られやすいことが知られている。また逆に、集中場所の 4 組の対応点の組を選ぶと対応点のある局所部分のみが一致し、その他の部分が全く一致していない、ということが多くなる。

そこで、処理時間の短縮のため、まず重なり部分を 4 つの領域に分割し、その領域から 1 点ずつ取りだしその中で最適値を求める、という手法を取った。

本研究で行ったアルゴリズムは以下のようなものである。

#### Step1

対応点を得られた時、全ての対応点を用いて対応点集合の重心を求める。

#### Step2

Step1 で求められた重心を分割位置として重なり部分を縦、横各 2 分割し、計 4 つの領域を得る。

#### Step3

式 (7) の最小値を保持しておく変数  $E$  を初期化する。

#### Step4

各領域から 1 組ずつ対応点組を抜きだし、平

面射影変換を求める。得られた平面射影変換を使って式 (7) の値を求める。

### Step5

$E$  よりも式 (7) の値が小さかったら  $E$  を現在の値に更新。まだ探索していない組があれば Step4 へ戻る。もう探索する組合せがない場合は、 $E$  の値を最小値とし、そのときの 4 組を最適な対応点の組として終了。

## 3.4 平面射影変換を用いたパノラマ画像生成

求められた平面射影変換を用いることにより、図 5 のように画像を重ね合わせ、パノラマ画像を生成することができる。基準画像上のある任意の画素が重ね合わせる画像上のどの画素に対応するかを式 (6) で求める。そして重ね合わせる画像上の輝度値を、対応する基準画像の画素の画素値とする。この作業を、基準画像の全画素に対して行うことにより、パノラマ画像を生成する。

### 3.4.1 基準画像に隣接していない画像のパノラマ画像生成

基準画像と隣接していない画像を同一画像上でモザイクするには、図 6 のように、基準画像に隣接している別の画像を介して平面射影変換パラメータを算出した。

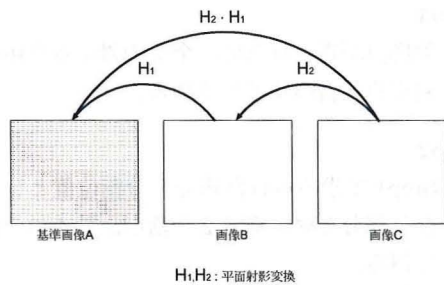


図 6: 間接的な平面射影変換

図 6 の場合、基準画像が画像 A であり、画像 B は A と隣接しており、 $H_1$  により A に重ねることができる。画像 C は B とは隣接しており  $H_2$  により

重ねることができる。このとき、以下の行列式 (8) により画像 C を A に重ねることが可能である。

$$A = H_1 \cdot H_2 C \quad (8)$$

## 3.5 パノラマ画像列を用いた形状復元

上記までの処理を行い、各カメラ位置においてモザイクを行い、パノラマ画像を生成した。そのパノラマ画像列を用いて、形状復元を行う。本研究で行うパノラマ画像列を用いた形状復元の手順を、図 7 に示す。

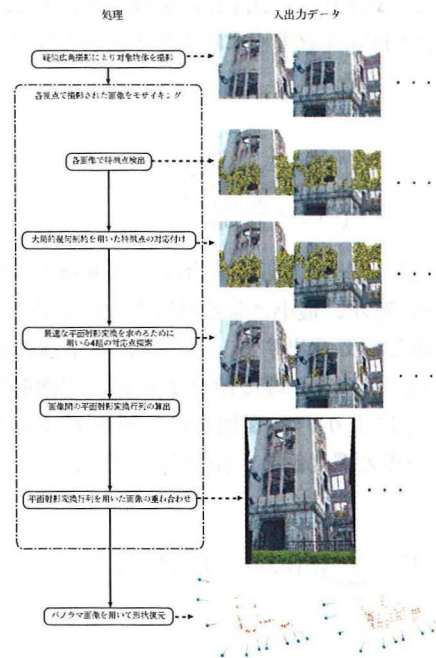


図 7: パノラマ画像列を用いた形状復元の手順

## 4 実験と評価

本手法を評価するために、まずモザイクを行う際に使用する対応点を求める手法の比較を行った。最適な対応点探索の高速化についての評価実験を行った。そして、モザイクを用いた形状復元に関する評価実験を、シミュレーションデータおよび実画像を用いて行った。ただし、各実験は、

Pentium4, OSはVine2.5, CPU:2GHzの計算機上で行った。

#### 4.1 モザイクングの評価

本手法で対応点探索に用いた大局的幾何制約を用いた特徴点の対応付け手法と、Zhangの手法および金谷の手法を同一画像に対して適用し、対応点検出の比較を行った。実験に用いたのは、図8の2枚であり、画像1を基準画像として対応点探索を行い、対応点を得た。このとき、各対応点の遷移を調べるため、オプティカルフローを作成して確認した。ここで、オプティカルフローとは、動画像における運動物体の見かけの速度場のことである。

そして、得られた対応点の組から、本手法で用いたモザイクング手法により各手法でモザイクングを行った。ここで、各手法で用いた画像の解像度は1024×768pixelである。

各手法で得られた対応点と、そのオプティカルフローを図8～11に示す。



図 8: 対応点検出に用いた画像

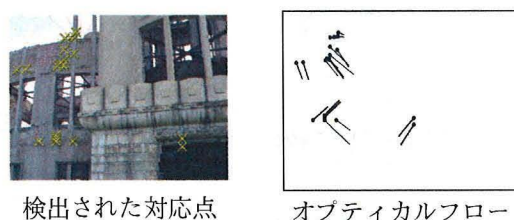


図 9: Zhangの手法による対応点検出結果

各手法で得られた対応点組から、3.3節で述べた手法で平面射影変換の算出を行い、実際にパノラマ画像を生成した結果を、図12に示す。

#### 考察

本手法とZhangの手法、金谷の手法の結果を比較する。Zhangの手法は、得られた特徴点数が少な

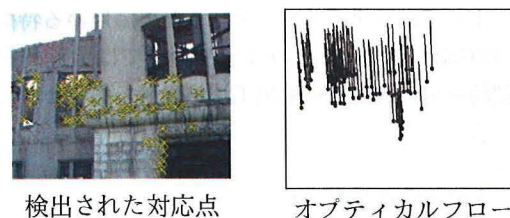


図 10: 金谷の手法による対応点検出結果

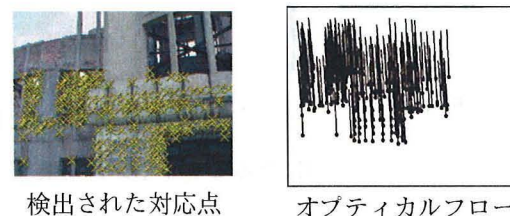


図 11: 大局的幾何制約を用いた手法による対応点検出結果

く、また、図9右図のオプティカルフローを見てわかるように線の方向がばらばらになっている。これは、対応点の遷移の方向がばらばらであることを示している。通常、オプティカルフローはほぼ一定方向へ遷移するはずであるから、この対応付けは誤対応を含んでいる。誤対応が含まれている組から平面射影変換を求めた場合、その誤対応点が重なり合うような平面射影変換が算出されてしまい、適切なモザイクングを行うことはできない。よって、この画像例に関していえば、モザイクングを行うために用いる特徴点の対応付けには、Zhangの手法から求められた対応点組を用いるのは適切ではないと考えられる。

一方、金谷の手法と本手法は、図10の右図および図11の右図を見るとわかるように、オプティカルフローがほぼ一定方向へのびているのがわかる。これは、特徴点がほぼ一定方向へ遷移していることを表している。一方、図10の左図および図11の左図より、本手法で対応付けした場合、重なり部分全体にまんべんなく対応点検出されているのに対し、金谷の手法では、重なり下部でほとんど対応点検出されていないことがわかる。ここで、平面射影変換を求める時、4組の対応点はできるだけ離れている方が適切な平面射影変換が算出されやすい。そのため、重なり部分全体から対応点組を選択できるほうが、平面射影変換を算出するときに有利である。



よって、モザイクングを行うために用いる特徴点の対応付けには、この3手法の中では大局的幾何制約を用いた対応点検出手法が最適であると考えられる。



Zhangの手法      金谷の手法      大局的幾何制約を用いた手法

図 12: 各手法によるモザイクング結果

## 4.2 高速化に関する実験

対応点の領域分割による高速化の効果を確認するため、同じ画像組を用いて、高速化を行わなかった場合と高速化を行った場合でそれぞれ平面射影変換を求める処理を行った。処理にかかった時間を表1に示す。また、モザイクングに用いた原画像を図13に示す。



基準画像      重ね合わせ画像

図 13: 平面射影変換算出に用いた画像

表 1: 平面射影変換算出にかかった時間

	usr(秒)	sys(秒)	real(秒)
高速化あり	66.860	0.180	1:08.43
高速化なし	613.380	0.480	10:29.95

高速化をせず得られた結果を、図14に、高速化を行なって得られた結果を図15に示す。

### 考察

表2をみると、高速化あり、なしともに同じ平面射影変換パラメータが求められていることがわかる。



図 14: 高速化なしから得られた結果      図 15: 高速化ありから得られた結果

パラメータの値が同じならば同様の重ね合わせが行われるので、図14、図15のようにモザイクング結果は等しくなる。しかし、処理時間をみると、高速化ありのほうが高速化なしに比べ10分の1程度に減少していることが分かる。

よって、この例では、高速化を行うことにより処理時間が短縮され、最適なパラメータを求めることができたといえる。

## 4.3 シミュレーション実験

パノラマ画像列を用いた形状復元の有効性を確かめるために、シミュレーション実験を行った。シミュレーション実験には、凹凸のある建物を想定し、球を変形させた図16のような7角形の形状データを対象物体として用いた。

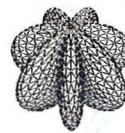


図 16: 対象物体

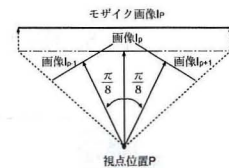


図 17: 疑似広角撮影法による画像生成

図16上の特徴点を1200点とし、10ヶ所から22.5度見上げる角度で画像を作成した。但し、同一視点Pからは図17のような疑似広角撮影法により3枚の画像  $I_{p-1}, I_p, I_{p+1}$  を撮影した。

### 通常の画像列を用いた形状復元

図17のような疑似広角撮影法により得られた3枚の画像の例を図18へ示す。

各画像の大きさは、 $600 \times 450$ pixelで、焦点距離は画像の横幅と同じとした。また、カメラ位置は、

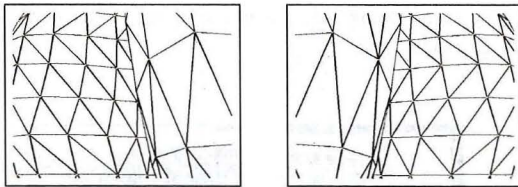
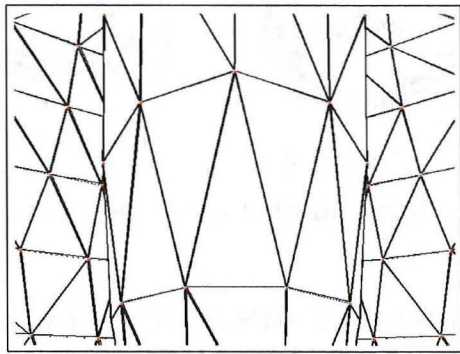


図 18: 疑似広角撮影により同一視点から得られた画像例

物体周辺を一回転する五角形の軌跡に沿って移動させた。

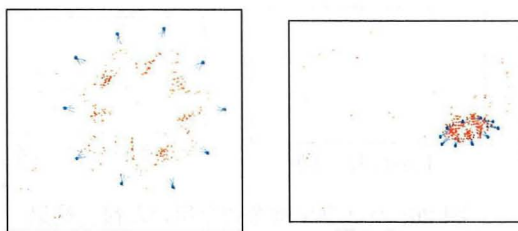


図 19: 通常の画像列を用いた復元結果

画像列から形状復元を行った結果、図 19 のような 3 次元形状が得られた。

#### パノラマ画像列を用いた形状復元

同一位置から撮影した 3 枚の画像をモザイクングしパノラマ画像を生成し、その画像列を用いて形状復元を行った。

図 18 の 3 枚の同一視点から撮影した画像をモザイクングして得られたパノラマ画像の例を、図 20 に示す。

このとき、画像枚数は 10 枚となり、各画像の大きさは、 $1384 \times 600$  pixel となった。他のパラメータは通常の画像列を用いた形状復元と同じとした。

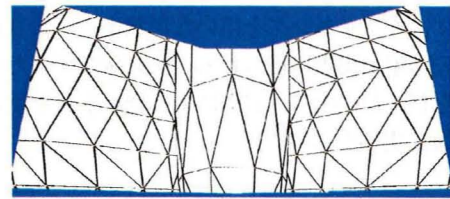


図 20: パノラマ画像例

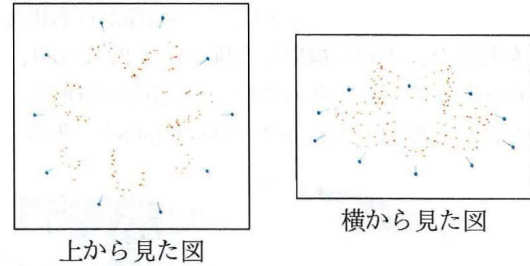


図 21: パノラマ画像列を用いた復元結果

図 21 が、パノラマ画像列から形状復元を行った結果である。

#### 考察

通常の画像列を用いた復元結果である図 19 を見ると、シミュレーションの形状データとは明らかに異なる位置に復元された点が多く存在していることがわかる。

一方、パノラマ画像列を用いた復元結果である図 21 を見ると、シミュレーションの形状データとほぼ同じ形状に復元されている、また、通常の画像列を用いた場合には大きな復元誤りが発生していたが、パノラマ画像列を用いて復元を行った結果には、明らかに誤っている位置に復元されている点は存在していない。

このように、疑似広角撮影法で撮影した画像に対して、通常の画像列を用いて形状復元を行うと、適切な形状に復元されないが、モザイクングを行い、生成されたパノラマ画像を用いて形状復元を行うことにより、適切な形状が復元される例が存在するということがわかった。

#### 4.4 実画像からの形状復元実験

実画像を用いて形状復元を行うとき、適切な 3 次元形状を得ることができるのかどうかを評価するた



めに、実画像を用いての形状復元の実験を行った。最初に、モザイクングを用いた形状復元により適切な復元結果が実際に得られるのかを確認するための実験を行った。次に、実際に本研究の手法を用いた形状復元実験を行った。

#### 4.4.1 パノラマ画像列を用いた形状復元

広島市立大学の講堂を 29 視点から疑似広角撮影した画像をモザイクングし、その画像から形状復元を行った。実際に撮影した画像は図 22 であり、この画像から手動で各画像間での対応点を検出した。ただし、画像の大きさは 540×360pixel である。

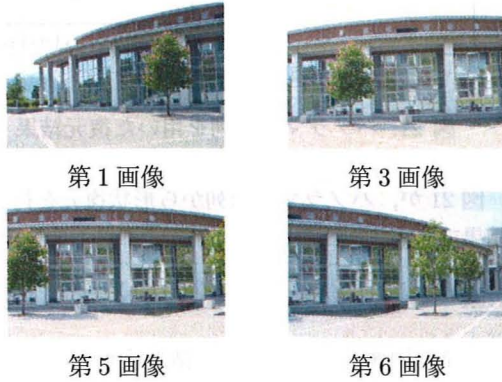


図 22: 同一視点からの撮影画像列

この画像列から 3 次元形状の一括復元を行った結果を図 23 に示す。また、図 23 の結果にテクスチャマッピングを行ったものを図 24 に示す。

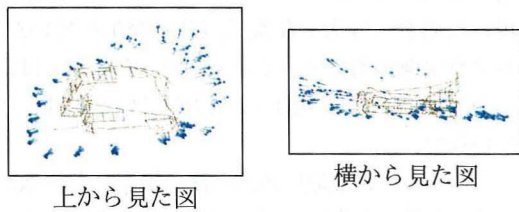


図 23: 通常の画像列を用いた復元結果

ここで各視点で撮影した画像にモザイクングを行い、1 視点につき 1 枚のパノラマ画像を作成し、その画像を使用して形状復元を行った。実際に作成し形状復元に用いたパノラマ画像例を図 25 に示す。ただし、この実験では、モザイクングを用いた形状復元の有効性を確かめるために、自動で対応点検索を利用したモザイクングを行わず、手動で特徴点の

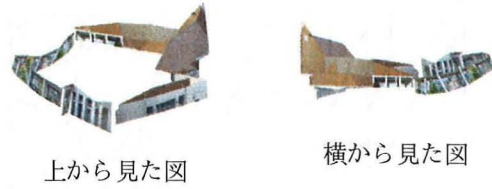


図 24: 通常の画像列によるテクスチャマッピング結果

対応付けを行いその情報をもとにモザイクングを行った。形状復元を行った結果を、図 26 に示す。また、図 26 の結果にテクスチャマッピングを行ったものを図 27 に示す。



図 25: パノラマ画像列

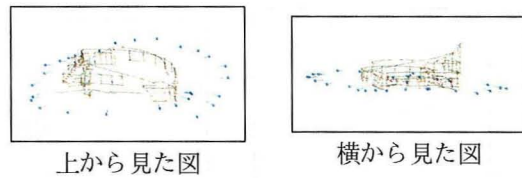


図 26: パノラマ画像列を用いた復元結果

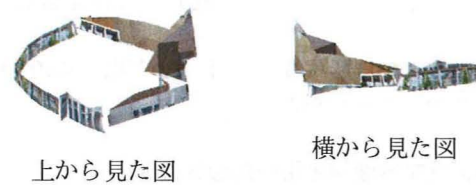


図 27: パノラマ画像列によるテクスチャマッピング結果

#### 考察

通常の画像列から形状復元を行ったとき、図 23 を見ると、カメラ位置が実際に撮影された位置とは異なった位置に復元されていた。また、本来扇状であるはずの右図の左から奥にかけての部分が大きく歪んでいるのがわかる。

一方、パノラマ画像列からの形状復元の結果である図 26 を見ると、図 23 では誤って復元されていた本来扇状である部分が適切に復元されていることがわかる。

よって、疑似広角撮影により撮影した画像から形状復元を行って適切な復元結果が得られなくても、画像に対しモザイクングを行い、生成されたパノラマ画像を用いて形状復元を行った場合、適切な復元結果を得られる例があることが確認できた。

#### 4.4.2 モザイクングを利用した形状復元の評価

本研究で提案した手法を用いて、実際に形状復元実験を行った。また、比較のために、通常の画像列を用いた形状復元も行った。ただし、復元の際に使用した対応点は、パノラマ画像を生成するときを使用した点ではなく、新たに手動で対応付けを行ったものである。

##### 通常の画像列を用いた形状復元

まず疑似広角撮影により広島市にある原爆ドームを 1 視点につき 3 枚ずつ 10ヶ所の視点から撮影した。撮影した 30 枚の画像を用いて、形状復元を行った。



図 28: 復元に用いた画像例

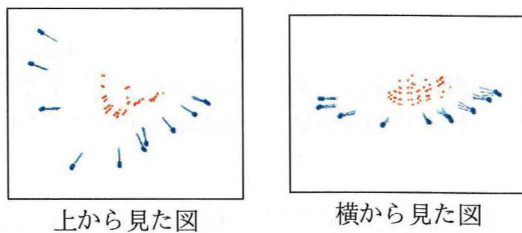


図 29: 復元結果

図 28 は、形状復元に使用した画像の一例である。ただし、画像の解像度は  $614 \times 460$  pixel であり、対応点は各画像から手動で検出した。

図 29 は、図 28 の画像列から抽出した対応点を元に復元を行った結果である。

##### パノラマ画像列を用いた形状復元

通常の画像列を用いた形状復元に用いた各視点での 3 枚の画像に対しモザイクングを行った。合計 10 枚のパノラマ画像列を生成し、その画像を用いて形状復元を行った。



図 30: 復元に用いた画像例

図 30 は、形状復元に使用したパノラマ画像の一例である。ただし、画像の解像度は  $780 \times 780$  pixel であり、対応点は全画像中から 56 組を検出した。

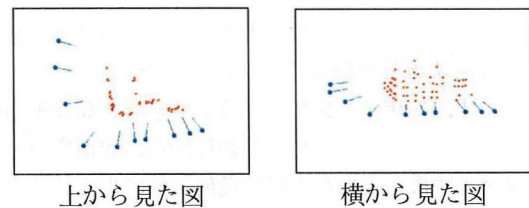


図 31: 復元結果

図 31 は、図 30 の画像列から手動でとった対応点を元に形状復元を行った結果である。

##### 考察

通常の画像列を用いて形状復元を行った結果である図 29 とパノラマ画像列を用いて形状復元を行った結果である図 31 を比較すると、パノラマ画像を用いた復元結果は、通常の画像列を用いて復元を行った結果とほぼ一致していることがわかる。

ここで、各復元結果の 2 乗平均平方根 (以下 RMS) について比較を行った。RMS とは、式 (9) の値のことであり、この式の値が小さい程、形状復元結果と、特徴点の誤差が小さいことを意味する。

$$\text{RMS} = \sqrt{\frac{\sum |\tilde{u}_{fp} - u_{fp}|^2}{n}} \quad (9)$$

ここで、 $n$ は用いた特徴点の数を表す。

本手法と従来手法のRMSを記述したものが、表2である。

表2: パノラマ画像列での復元結果と通常の画像列での復元結果のRMS

	画像枚数(枚)	RMS(pixel)
パノラマ画像列からの復元	10	0.768840
通常の画像列からの復元	30	1.137106

表2を見ると、モザイク画像を用いた本手法の方がRMSが小さいことが分かる。よって、RMSという視点から判断した場合、パノラマ画像列から形状復元を行った方が適切な結果が得られたと考えられる。

## 5 おわりに

本研究では、同一視点から撮影された画像をモザイクキングすることにより、1枚の画像に対象物体全体が含まれるパノラマ画像を生成し、パノラマ画像から形状復元を行う手法を提案した。

まず、モザイクキングを行う上で必要となる平面射影変換を算出するために各画像から、並進変換とアフィン変換を用いた大局的幾何制約による特徴点の対応付けを行った。次に、処理を高速化するために、対応付けで得られた対応点全体の重心をとり、その重心を分割点として4つの領域に対対応点を分割した。そして、その4領域の中から一点ずつを選択し適切なモザイクキングが行える平面射影変換を算出することができる4組の対応点の探索を行った。そして、求められた対応点から平面射影変換パラメータを算出し、この平面射影変換パラメータを用いてパノラマ画像を作成した。最後に、このパノラマ画像列を用いて形状復元を行った。

実験においては、まずモザイクキングのために用いる対応付け手法として、本手法で用いた大局的幾何制約を用いた特徴点の対応付け手法が適切であるかどうかの評価を行った。大局的制約を用いた対応付け手法と、特徴点の対応付け手法として一般的なZhangの手法、金谷の手法との比較を行なった。その結果、本手法はモザイクキングを行うための対応付け手法として適切であるということが確認できた。

次に、本研究で行った、対応点の領域分割による処理の高速化と適切な平面射影変換パラメータの算出が行われているかどうかを確認するために実験を行った。その結果、処理時間は短縮され、また、完全探索を行ったときと同じ値が算出された。よって、領域分割による高速化が行えていることを確認した。

次に、パノラマ画像列を用いて形状復元を行う有効性を評価するために、通常の画像列を用いた場合形状復元が適切に行えないシミュレーション画像と実画像に対して実験を行った。その結果、シミュレーション画像、実画像での結果はどちらもモザイクキングを行うことにより適切な形状復元結果が得られた。

今後の課題としては、モザイクキングがどの程度有効であるのか、評価尺度を定義してシミュレーション実験により定量的に評価することが挙げられる。また、形状復元に関してはすべて手動で行っていたが、文献[2]の手法と統合し、形状復元を自動化することが挙げられる。さらに、今回はモザイクキングを行うときに使用した特徴点と形状復元を行うときに使用した特徴点をそれぞれの処理で別々に検出しているが、同じ画像であることから、モザイクキング処理で検出した特徴点に対しモザイクキングで求められた平面射影変換により座標を移動させて形状復元を行う際の特徴点として用いる特徴点の再利用が挙げられる。

## 参考文献

- [1] Oishi, T., Masuda, T. and Ikeuchi, K.: "Digital Restoration of the Cultural Heritages," *Proc.VSMM*, pp. 934-941, 2002.
- [2] 保田和隆, 右田剛史, 青山正人, 椋木雅之, 浅田尚紀: "疎な全周囲画像列からの密な3次元形状モデルの生成", 情報研報, 2003-CVIM-138, pp. 73-80, 2003.
- [3] 椋木雅之, 右田剛史, 天野晃, 青山正人, 浅田尚紀: "再投影誤差最小化による建物画像列からの全周形状の一括復元", 情報研報, CVIM-142, pp.81-88, 2004



- [4] 千葉直樹, 蚊野浩, 美濃導彦, 安田昌司: “画像特徴に基づくイメージモザイク”, 信学論, J82-D-II, pp.1581-1589, 1999
- [5] R. Szeliski., “Image Mosaicing for Tele-Reality Applications,” Technical Report 94/2, Digital Equipment Corporation, Cambridge Research Lab, 1994.
- [6] I. Zoghlami., O. Faugeras., and R. Deriche., “Using Geometric Corners to Build a 2D Mosaic from a Set of Images,” *IEEE Proc. Computer Vision and Pattern Recognition*, Tracked Feature Points with Fast Nonlinear Optimization, pp. 420-425, 1997.
- [7] Z. Zhang, R. Deriche, O. Faugeras and Q.-T. Luong : “A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry,” *Proc. 3rd Artif. Intell.*, **78**, pp. 87-119, 1995
- [8] 金澤靖, 金谷健一: “段階的マッチングによる画像モザイク生成”, 信学論, J86-D-II, pp. 816-824, 2003.
- [9] 金澤靖, 金谷健一: “コンピュータビジョンのための画像の特徴点抽出”, 信学論, Vol. 87, No. 12, pp. 1043-1048, 2004.
- [10] C. Harris and M. Stephens: “A Combined Corner and Edge Detector,” *Proc. Alvey Vision Conf.*, pp. 147-151, 1988.
- [11] P. Beardsley, P. Torr and A. Zisserman: “3D Model Acquisition from Extended Image Sequences,” *Proc. 4th ECCV'96*, Cambridge, UK., vol. 2, pp. 683-695, 1996.
- [12] Reinhard Koch, Marc Pollefeys and Luc Van Gool: “Multi Viewpoint Stereo from Uncalibrated Video Sequences,” *Proc. ECCV'98*, LNCS, Springer-Verlag, Freiburg, pp. 55-71, 1998.
- [13] 佐藤智和, 神原誠之, 横矢直和, 竹村治雄: “マルチベースライン法を利用した動画からの野外環境の三次元モデル化”, 信学技報, PRMU2001-233, pp. 1-8, 2002.
- [14] M. Lhuillier and L. Quan: “Quasi-Dense Reconstruction from Image Sequence,” *ECCV'02*, pp. 119-139, 2002.
- [15] 中川雅朗, 斎藤英雄, 小沢慎治: “ハンディカメラを用いて撮影された画像列からのEPIの補正に基づく室内環境3次元再構築”, 信学論, Vol. J84-D-II, pp. 266-275, 2002.
- [16] A. Amano, T. Migita and N. Asada: “Stable Recovery of Shape and Motion from Partially Tracked Feature Points with Fast Nonlinear Optimization,” *Proc on Vision Interface*, pp. 244-251, 2002.
- [17] 星野准一: “1次元フローに基づく高速なパノラマ画像の生成”, 画電学誌, Vol. 27, No. 4, pp. 396-404, 1999.
- [18] 星野准一, 増田 功: “局所的な輝度変動にロバストなパノラマ画像生成法”, 信学論, Vol. J82-D-II, No. 2, pp. 222-229, 1999.
- [19] 谷口行信, 阿久津明人, 外村佳伸: “Panorama Excerpts: パノラマ画像の自動生成・レイアウトによる映像一覧”, 信学論, Vol. J82-D-II, No. 3, pp. 390-398, 1999.
- [20] Y. Kanazawa, K. Kanatani, “Image mosaicing by stratified matching,” *Image and Vision Computing*, Vol. 22, pp. 93-103, 2004.

# 線分要素に基づく画像列からの 物体形状モデルの生成

## 1 はじめに

実物体の3次元モデル化は、近年普及してきたインターネットでのリアルな仮想空間の構築や、デジタルアーカイブにより保存された芸術作品や文化財のウェブ上での公開など多様な目的に用いられている重要な技術である [1]。3次元モデルの生成のためには、レーザレンジセンサのような専用の計測装置を用いて高精度な3次元モデルを生成する手法がある。しかし、この手法には多大なコストと時間を費やしてしまい、そして対象物体は大きさや表面特性などいくつかの撮影条件を満たさなくてはならないという問題がある [2]。したがって従来より、迅速かつ手軽に3次元形状を生成できるシステムが必要とされ、コンピュータビジョン技術の応用が研究されている。その1つとして、複数の視点から対象物体を撮影した画像(多視点画像)より、その3次元形状を復元する手法が挙げられる。この手法には特殊な計測装置は不要であり、対象物体の大きさが制限されることもないため低コストで実現ができる。そのため、画像列からの3次元形状復元の研究が数多く行われている [3]。

画像列からの形状復元では、視点の異なる各画像間において対応した特徴点の探索を行う必要がある。この特徴点の対応付けを正確に行うことができれば、透視投影のモデルに基づき各点の3次元上の位置の復元が可能となる。そのため、特徴点の追跡や対応付けの手法 [4, 5, 6] が提案されている。また、カメラ位置・姿勢が既知であれば、マルチベースラインステレオ法 [7] や距離推定 [8]、多視点解析 [9] などにより、特徴点群からなる部分モデルの生成が可能である。また、大局的な幾何制約を用いることで信頼性の高い特徴点の対応付けを可能とし、段階的にモデルの精度向上を行う手法 [3] が提案されている。しかし、直線的な部分においては特徴点の対応付けの精度が低く、特徴点の追跡結果から得られる形状復元が不安定となるため、生成されるモデルの線分要素に歪みが生じてしまうという問題がある。

そこで本研究では、直線的な要素で構成された人工物を対象として、物体の全周囲を撮影した画像列から、直線状の物体輪郭線で表された3次元形状を復元し、そのモデルの輪郭線を高精度に生成することを目的とする。特に、明確な線分で構成された3次元モデルを生成するために、特徴点を用いた形状復元ではなく、復元物体のエッジに基づく特徴線を用いた形状復元を行う。特徴線とは、直線方向に1次元の自由度をもつ特徴点のことであり、同一直線上であればどの位置に与えてもよい。したがって、厳密な1点を与えなければならない特徴点に比べて、対応付けによる形状復元の安定化が期待できる。本研究は文献 [10] の手法を拡張したものであり、特徴線の端点を検出することで、特徴線を用いた形状復元から3次元モデルを生成することを可能としている。なお形状復元は、特徴線が画像全体で追跡されている必要は無く、追跡誤差を補正できる非線形最適化を用いて行う。

本手法では、まず、各画像において物体のエッジを検出し、エッジ上に手で特徴線を与える。特徴線は物体の直線状構造に対して与えるので、手動であっても対応付けは容易となる。次に、対応付けた特徴線による形状復元を行い、各交点を計算する。そして、特徴線の端点として適切な交点を判定する。ここで、端点をもつ特徴線を物体の線分要素とみなし、その端点を物体の頂点として扱う。求めた頂点と線分要素の連結関係により3角面を定義することで、3次元モデルの生成を行う。本研究で最終的に得られる3次元モデルは、各面が3角面で定義される多面体で構成されており、各3角面には撮影画像によるテクスチャマッピングが行われている。

本論文では、まず2節で、画像系列からの形状復元を行うための一般的手法や従来手法を示す。次に、3節では線分要素に基づく3次元モデルを生成する手法について述べる。そして、4節で本研究の適用結果をシミュレーション画像列と実画像列を用いて示す。最後に5節をまとめとする。

## 2 画像列からの3次元形状復元の原理

### 2.1 特徴線を用いた3次元形状復元

文献 [10] では、特徴点の代わりに特徴線を形状復元のために用いる。次節以降で特徴線を用いた形状復元の処理について詳細に述べる。

#### 2.1.1 特徴線

特徴線は3次元空間中の直線であり、投影された特徴線は画像列上でも観測される。第  $f$  画像、第  $p$  特徴線の投影は画像中心  $c$  を原点  $c(0,0)^T$  とする画像座標系  $uv$  上の位置  $u_{fp} = (u_{fp}, v_{fp})^T$  と方向ベクトル  $a_{fp} = (a_{fp}, b_{fp})^T$  で表現する (図 1)。なお、本研究では各画像に特徴線を入力して与えることを特徴線の対応付けと呼ぶ。

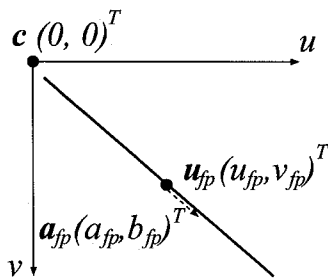


図 1: 投影された線特徴

特徴線は、基本的に特徴点に方向ベクトルを付加させた”方向付き特徴点”と捉えることができ、直線として扱う。特徴線は方向ベクトル  $a$  が一定かつ  $u' = u + ha$  ( $h$  は任意の実数) であれば、 $u, u'$  のどちらも同じ特徴線上の位置として扱ってよい (図 2)。

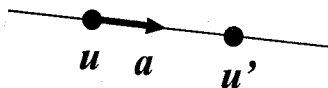


図 2: 同一の特徴線

#### 2.1.2 対応付けと補正

##### 2次元 Laplacian of Gaussian フィルタ

画像中の直線成分に特徴線の対応を与えるために、エッジ検出を行った画像を用いる。エッジとは、画像内で明るさやその空間微分に急激な変化が起こる境界線である。ここではエッジの強度は明度で表現する。明度の変化は、画像に投影された物体表面の重要な特徴を反映するものである。エッジが現れるのは、例えば以下に示すような場合がある。

- 物体表面の方向が不連続に変化する
- 物体が他の物体を遮蔽する
- 影の境界線が現れる
- 物体表面の反射率特性の変化が不連続な領域である

黒澤の手法ではガウス分布の2次微分である  $\nabla^2 G$  (2次元 Laplacian of Gaussian, 2次元 LoG)[14] をフィルタとして用いた。 $\nabla^2 G$  は式 (1) で表される。

$$\nabla^2 G(x, y) = \frac{x^2 + y^2 - 2\sigma^2}{2\pi\sigma^6} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (1)$$

式 (1) の  $\sigma$  によってエッジの検出感度を制御することができる。 $\sigma$  を小さくした場合、平滑化される度合いが少なくなり、弱いエッジも検出できる。 $\sigma$  を大きくした場合には、逆に強いエッジのみが検出されるようになる。また、1次元の微分演算子は方向性があり、得られる結果はエッジの方向に依存するが、2次元の微分演算子ではすべての方向のエッジを対等に扱える。したがって、この2次元 LoG フィルタを用いることで全方向のエッジを同時に見つけることができる。実際には、 $\sigma = 0.25$  として、フィルタサイズが  $19 \times 19$  の LoG フィルタを用いる。

#### 特徴線の入力

文献 [10] では、特徴線の初期位置を手動で入力し、その近傍領域から位置と方向を自動抽出する。図 3 はあるエッジ画像を拡大したものであり、強いエッジ部分ほど暗い灰色で表示している。この画像中には右上から左下にかけて直線に分布するエッジがあり、これを特徴線とみなし、付近に手動で特徴線の

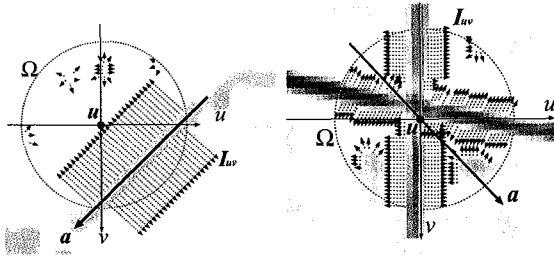


図 3: 特徴線の入力

図 4:  $\Omega$  内の 2つのエッジの直線成分

初期位置  $u$  を与える。次に、初期位置  $u$  を中心とする近傍の円領域  $\Omega$  を考える。 $\Omega$  内の画素について  $u$  方向に隣接する画素の明度の差分  $I_u(u, v)$ ,  $v$  方向に隣接する画素の明度の差分  $I_v(u, v)$  を計算する。この勾配ベクトル  $\mathbf{I}_{uv}(u, v) = [I_u(u, v), I_v(u, v)]^T$  はエッジの明度について、勾配の向きと変化量を表す(図 3 の破線矢印)。 $I_{uv}(u, v)$  はエッジの直線成分上では、その直線に対する法線ベクトルと解釈でき、エッジの直線成分上ではほぼ一定の方向を示す。 $\Omega$  内の全  $m$  画素の勾配ベクトルより  $\mathbf{I}'_{uv}$ (式 (2)) が得られる。 $\mathbf{I}'_{uv} \mathbf{I}'_{uv}^T$  の最小固有ベクトルを特徴線方向ベクトルとする。

$$\mathbf{I}_{uv} = \begin{bmatrix} I_u(u, v) \\ I_v(u, v) \end{bmatrix} = \begin{bmatrix} I(u+1, v) - I(u, v) \\ I(u, v+1) - I(u, v) \end{bmatrix}$$

$$\mathbf{I}'_{uv} = \begin{bmatrix} \mathbf{I}_{uv}^T(u_0, v_0) & \dots & \mathbf{I}_{uv}^T(u_n, v_n) \end{bmatrix}^T \quad (2)$$

$\Omega$  には図 3 の左上にある短い破線矢印で示しているノイズとなるエッジも存在するが、大半が特徴線上にあるエッジとみなし、無視する。しかし、図 4 のように 2つのエッジの直線成分付近に特徴線的位置を与えた場合、それぞれの直線成分を 1つとみなして特徴線方向を計算するため、どちらの方向ベクトルも求めることができない。図 4 は直線成分の交点であり、特徴点は従来この交点の位置に与えることがほとんどである。そのため、特徴点と同じ位置に特徴線を与えることはできない。よって、特徴線の初期位置は半径 10pixel の  $\Omega$  内にエッジの直線成分が 1つのみある位置を与えるものとする。

## モーメントによる補正

手で与えた特徴線の初期位置は正確ではないため、位置の補正を行う。特徴線は線上に自由度を持つので、特徴線的位置を法線方向のみ補正し、それに伴い特徴線方向も 2.1.2 節の手法で再計算する。

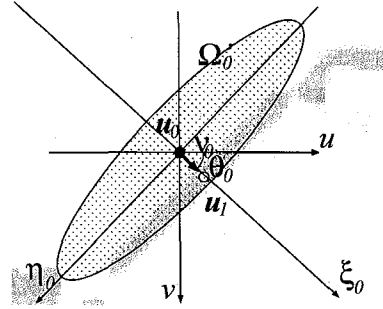


図 5: 楕円の近傍領域と特徴線の補正

この補正にはまず、特徴線を入力したときに求められる方向ベクトルを軸とする  $\eta$  から垂直ベクトルを軸とする  $\xi$  を求める。そして、図 5 のように  $\xi$  方向に領域  $\Omega_0$  内エッジ強度のモーメントが 0 になる(式 (3)) ような移動量  $\nu_0$ (式 (4)) を計算する。なお、 $\xi$  軸は  $u$  軸を  $\theta_0$  回転したものである。移動後の特徴線位置を  $u_1$  として特徴線方向を計算し、再びモーメント 0 になるように特徴線の垂直方向へ修正する。最終的に修正量  $\nu = 0$  になった位置を最終結果とする。

$$\iint_{\Omega'} I(\xi)(\xi - \nu) d\xi = 0 \quad (3)$$

$$\nu = \frac{\iint_{\Omega'} \xi \cdot f(\xi) \cdot d\xi}{\iint_{\Omega'} f(\xi) d\xi} \quad (4)$$

$uv$  座標系で考えると、特徴線の法線角度  $\theta_0$ , 回転行列  $\mathbf{R}_0$  とし、

$$\mathbf{R}_0 = \begin{bmatrix} \cos \theta_0 & -\sin \theta_0 \\ \sin \theta_0 & \cos \theta_0 \end{bmatrix}$$

$$\nu_0 = \frac{\sum \xi \cdot I \left( \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} + \mathbf{R}_0 \begin{bmatrix} \xi_0 \\ \eta_0 \end{bmatrix} \right)}{\sum I \left( \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} + \mathbf{R}_0 \begin{bmatrix} \xi_0 \\ \eta_0 \end{bmatrix} \right)}$$

補正位置  $\mathbf{u}_1$  は

$$\mathbf{u}_1 = \begin{bmatrix} u_0 + \nu_0 \sin \theta_0 \\ u_0 + \nu_0 \cos \theta_0 \end{bmatrix}$$

となる。補正後の位置  $\mathbf{u}_1$  から特徴線の法線方向を再計算し、新たな近傍領域を求めて  $\nu_n = 0$  になるまで補正を  $n$  回繰り返す。

$\Omega$  の面積を大きくすると、それに比例して補正に使用する画素情報が増すので、補正精度は向上すると考えられるが、特徴線以外のノイズも混入してしまう。よって、近傍領域  $\Omega'$  を特徴線の法線方向を短軸とする楕円にする。これにより、ノイズとなる特徴線上以外のエッジの影響を抑える。

この補正の精度を向上させるために、各特徴線について  $\Omega$  の長径と短径の比率を楕円円周上のエッジ情報から最適化することも考えられる。しかし、本研究では全ての特徴線について長径 : 短径 = 2 : 1 としている。

### 2.1.3 3次元形状復元

3次元形状復元とは、各特徴線の3次元空間での座標  $\mathbf{s}_p$  と方向ベクトル  $\mathbf{A}'_p = (x'_{fp}, y'_{fp}, z'_{fp})$  を計算により求めることである。このとき、各方向ベクトルは正規化しているものとする。

ここで、3次元空間にある特徴線を画像に投影した特徴線上の点の座標を  $\tilde{\mathbf{u}}_{fp} = [\tilde{u}_{fp}, \tilde{v}_{fp}]^T$  とする。このとき、方向ベクトル  $\mathbf{a}$  はそれ自身が位置の自由度を持つため、特徴線を表すために必要な  $\tilde{\mathbf{u}}_{fp}$  の座標は一意に決定できないが、本研究では線分の midpoint とする。このようなデータに対し、特徴点による3次元形状復元の手法をそのまま用いることは、特徴線が許す方向ベクトル  $\mathbf{a}$  の向きの位置のズレも投影誤差として含んでしまい、好ましくない(図6(左))。そこで、特徴点の最適化計算を行う際、方向ベクトル  $\mathbf{a}$  に沿った方向へのズレを許容するように重み付けを施す(図6(右))。実際には式(6)に示す重み付けを用いて式(5)を最適化することで計算を行う。

$$\arg \min_{\mathbf{s}_p, \mathbf{R}_f, \mathbf{t}_f} \sum_{f,p}^{F,P} \mathbf{h}_{fp}^T \mathbf{W}_{fp} \mathbf{h}_{fp} \quad (5)$$

ただし、 $\mathbf{W}_{fp}$ ,  $\mathbf{h}_{fp}$  はそれぞれ次式を満たす。

$$\begin{cases} \mathbf{W}_{fp} = [\mathbf{I} - \mu \mathbf{a}_{fp} \mathbf{a}_{fp}^T] & (0 \leq \mu \leq 1) \\ \mathbf{h}_{fp} = \mathbf{u}_{fp} - \tilde{\mathbf{u}}_{fp} \end{cases} \quad (6)$$

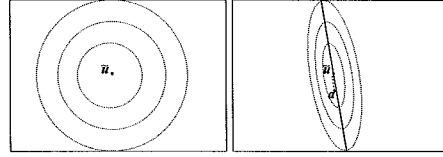


図6:  $\mathbf{u}$  の不確かさの分布 (左:方向無, 右:方向有)

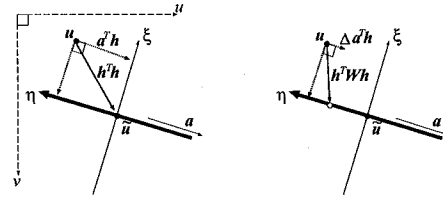


図7: 重み付けによる方向の2乗距離軽減 (左:重み付け無, 右:重み付け有)

式(5)に用いた重み  $\mathbf{W}_{fp}$  は推定する特徴線上の点  $\mathbf{u}_{fp}$  と画像上の点  $\tilde{\mathbf{u}}_{fp}$  の差  $\mathbf{h}_{fp} = \mathbf{u}_{fp} - \tilde{\mathbf{u}}_{fp}$  を方向ベクトル  $\mathbf{a}$  とそれに垂直な方向に分解して考えることにより導出することができる。このとき、画像座標系  $wv$  ではなく  $\mathbf{a}_{fp}$  を軸とし、 $\tilde{\mathbf{u}}$  を原点とする座標系  $\xi\eta$  を考える。

$\xi\eta$  座標系において  $\mathbf{u}$  と  $\tilde{\mathbf{u}}$  の差を  $\mathbf{h} = [h_\xi, h_\eta]^T$  とする。 $\xi\eta$  座標系は  $wv$  座標系を回転しただけなので、2乗距離  $\mathbf{h}^T \mathbf{h}$  は次式のように等しくなる(図7(左))。

$$h_\eta^2 + h_\xi^2 = h_u^2 + h_v^2 = \mathbf{h}^T \mathbf{h} \quad (7)$$

$\mathbf{a}$  は正規化されているので、回転行列として扱える。このとき、 $\eta$  方向についての残差は次式のようにになる。

$$h_\eta = \mathbf{a}^T \mathbf{h}. \quad (8)$$

次に  $\mathbf{a}$  の方向についてのズレが減少するように  $0 < \Delta < 1$  の範囲を取る重み付けを行う(図7(右))。

$\Delta^2 = 1 - \mu$  とすると評価式は式 (7), (8) より, 次式が求まる.

$$\begin{aligned} (1 - \mu)h_\eta^2 + h_\xi^2 &= h_u^2 + h_v^2 - \mu \cdot h_\eta^2 \\ &= \mathbf{h}^\top \mathbf{h} - \mu (\mathbf{a}^\top \mathbf{h})^2 \\ &= \mathbf{h}^\top \mathbf{h} - \mu \mathbf{a} \mathbf{h}^\top \mathbf{a}^\top \mathbf{h} \\ &= \mathbf{h}^\top [\mathbf{I} - \mu \mathbf{a} \mathbf{a}^\top] \mathbf{h} \quad (9) \end{aligned}$$

また, 定義した  $\mu$  は, 特徴線の伸縮性を表す重み係数であり,  $\Delta$  の定義に従って  $0 < \mu < 1$  となる.  $\mu$  が 1 に近いほど, 方向ベクトル  $\mathbf{a}$  の方向のズレを無視するようになる. また,  $\mathbf{W}_{fp}$  を単位行列 ( $\mu = 0$ ) に近づけると特徴点の推定となるので, 特徴線と特徴点は同時に扱うこともできる. 本研究では一貫して  $\mu = 0.99$  として手法を適用することにする.

次に, 形状復元された特徴線の 3 次元上の方向ベクトルを求める. まず, 視点から画像上に特徴線に対応付けた位置  $\mathbf{u}_{fp}$  に向かう方向ベクトルを  $\mathbf{U}_{fp} = (u_{fp}, v_{fp}, f)$  とする. 次に,  $\mathbf{a}_{fp}$  と  $\mathbf{U}_{fp}$  のベクトルからなる平面を考える (図 8).

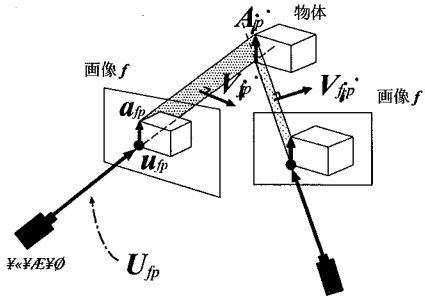


図 8: 方向ベクトルにおける画像座標系と世界座標系の関係

その平面の法線ベクトル  $\mathbf{V}_{fp}$  は外積を用いて次式のように表される.

$$\mathbf{V}_{fp} = \mathbf{a}_{fp}^\top \times \mathbf{U}_{fp} \quad (10)$$

この法線ベクトル  $\mathbf{V}_{fp}$  は画像  $f$  のカメラ座標系での方向ベクトルとなる. ここで, 次式のように世界座標系における方向ベクトル  $\mathbf{V}'_{fp}$  に変換する.

$$\mathbf{V}'_{fp} = \mathbf{V}_{fp}^\top \mathbf{R}_f^{-1} \quad (11)$$

3 次元上の特徴線の方向ベクトル  $\mathbf{A}'_p$  は全ての  $\mathbf{V}'_{fp}$  と直交している. したがって, 内積  $\mathbf{V}'_{fp} \cdot \mathbf{A}'_p = 0$  を

満たす. しかし, 実際には,  $\mathbf{V}'_{fp}$  には誤差が含まれるため, 次式により  $\mathbf{A}'_p$  を求める. なお,  $|\mathbf{A}'_p| = 1$  である.

$$\arg \min_{\mathbf{A}'_p} \sum_f \mathbf{V}'_{fp} \cdot \mathbf{A}'_p \quad (12)$$

なお, 行列  $\mathbf{V}'_p$  を次のように置く.

$$\mathbf{V}'_p = \begin{bmatrix} \mathbf{V}'_{1,p}^\top & \mathbf{V}'_{2,p}^\top & \dots & \mathbf{V}'_{F,p}^\top \end{bmatrix} \quad (13)$$

式 (13) を用いることで, 式 (12) は次式で表すことができる.

$$\arg \min_{|\mathbf{A}'_p|=1} \mathbf{A}'_p (\mathbf{V}'_p \mathbf{V}'_p^\top) \mathbf{A}'_p \quad (14)$$

したがって, 3 次元上の特徴線の方向ベクトル  $\mathbf{A}'_p$  は,  $\mathbf{V}'_p \mathbf{V}'_p^\top$  の最小固有値に対応する固有ベクトルとして求めることができる.

## 2.2 従来手法の問題点と本研究での解決法

以上の処理により, 画像列から形状復元を行い, 復元された形状を表現することができる. しかし, 多視点から撮影された実画像列を用いて全周囲 3 次元モデルを生成するには, 以下に述べる問題がある.

特徴点による形状復元から 3 次元モデルを生成する従来手法として, 文献 [3] の手法がある. この手法では, 対象物体の形状によっては形状復元が不安定となり, 3 次元モデルの生成が困難となる場合がある. 特に, 物体の線分要素では歪みが生じることが多く, 視覚的に好ましくない精度の低い結果になってしまう. 線分要素に歪みが生じることなく 3 次元モデルを生成するためには, 物体の線分要素上に特徴点が密に検出される必要がある. しかし, Harris の特徴点検出法では, 線分において輝度値が変化する領域が一様, あるいは 1 方向であることが多いため, 特徴点を多く検出することが困難となる. そのため, アフィン変換を用いた特徴点の対応付けと追跡結果による形状復元が不安定となり, それ以降の結果に影響を及ぼすことになる.

そこで本研究では, 直線的な構造をもつ人工物を対象として取り上げ, 線分要素に歪みのない 3 次元モデルを生成する. そのために, 線分要素の形状復元には特徴点ではなく, 特徴線を利用することで対

応付けを行い、物体を形状復元する。特徴線の形状復元については2.1節で説明した文献[10]の手法を適用して行う。画像から検出したエッジ強度の高い位置に手動で特徴線に対応付ける。特徴線の方向はエッジ強度に依存するが、対応付けた位置、方向ベクトルは修正される。したがって、特徴線による形状復元は安定した形状が期待できる。また、特徴線は特徴点に方向ベクトルを加えたものであるから、特徴点としても形状復元が可能である。一方、この手法では、特徴点の端点検出は行っていないため、テクスチャを張りつけた3次元モデルの生成ができない。

以上を踏まえて、線分要素が安定した3次元モデルを生成するために、本研究では、まず、検出されたエッジに特徴線に対応付ける。次に、形状復元された3次元特徴線の各交点を計算し、特徴線の端点を判定する。この際、端点をもつ特徴線を物体の線分要素とみなす。線分要素の端点を物体の頂点とし、線分要素との連結関係を求める。そして、得られた連結関係をもとに、頂点で構成された3角面を生成する。最後に、各3角面に対して適切なテクスチャを選択し、3次元モデルを生成するという手法を提案する。

### 3 線分要素に基づく3次元モデルの生成

#### 3.1 特徴線の交点

特徴線を復元することで対象物体の輪郭線を表示することができる。特徴線の交点をテクスチャの頂点とすることで、テクスチャを貼り付けた結果を出力することができる。

そこで、形状復元された特徴線から交点を求めることで、頂点として特徴線の端点を求めることになる。実際には交わるはずの線がねじれの位置にある場合を想定し、復元された2特徴線の距離の中点を交点として次のように求める。

図9のように点 $s_i$ を通り方向ベクトル $A_i$ の直線 $l$ 、点 $s_j$ を通り方向ベクトル $A_j$ の直線 $m$ は次式で表される。

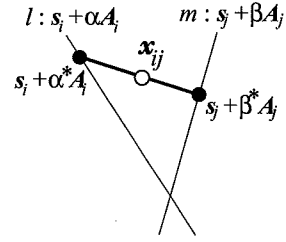


図9: ねじれの位置のある2直線の最も近い距離にある点の中点

$$\begin{aligned} l: & \quad s_i + \alpha A_i \\ m: & \quad s_j + \beta A_j \\ & \quad (\alpha, \beta \text{は実数}) \end{aligned}$$

2直線が最も近いときの各線上の点は、次の $\alpha^*, \beta^*$ を満たす。

$$\begin{aligned} s_i + \alpha^* A_i & \approx s_j + \beta^* A_j \\ (s_i - s_j) + (A_i | - A_j) \begin{bmatrix} \alpha^* \\ \beta^* \end{bmatrix} & \approx 0 \\ \begin{bmatrix} \alpha^* \\ \beta^* \end{bmatrix} & \approx \\ & ((A_i | - A_j)^T (A_i | - A_j))^{-1} (A_i | - A_j)^T (s_i - s_j) \end{aligned} \quad (15)$$

直線 $l, m$ の距離の端点はそれぞれ $s_i + \alpha^* A_i, s_j + \beta^* A_j$ となり、中点 $x_{ij}$ は以下のようにして求まる。

$$\begin{aligned} x_{ij} & = \frac{(s_i + \alpha^* A_i) + (s_j + \beta^* A_j)}{2} \\ & = \frac{s_i + s_j}{2} + \frac{1}{2} (A_i | A_j) \begin{bmatrix} \alpha^* \\ \beta^* \end{bmatrix} \\ \text{式(15)を代入して,} \\ & = \frac{s_i + s_j}{2} + \\ & \quad \frac{1}{2} (A_i | A_j) ((A_i | - A_j)^T (A_i | - A_j))^{-1} (A_i | - A_j)^T (s_i - s_j) \end{aligned} \quad (16)$$

この式(17)で表された中点 $x_{ij}$ が2つの特徴線の交点となる。

#### 3.2 線分要素と接続関係の検出

3.1節で求めた3次元上の交点から各特徴線の端点を判定し、線分要素とその接続関係を求める。こ

ここで、線分要素の端点を物体の頂点として扱う。ただし、頂点は3つの線分要素が1端点で接続しているものとする。以下にアルゴリズムを述べる。

1. 3次元形状を構成している全3次元線群から、1直線を取り出す。
2. 取り出した直線*i*に対して交差する直線*j*との方向ベクトルの絶対値差分が  $|A_i - A_j| \approx 0$  となる場合、つまり2直線*i*, *j*がほぼ平行であるとき、接続関係はなく、交点  $x_{ij}$  は端点として考慮しない。
3. 直線*i*と交点  $x_{ij}$  との距離が閾値よりも小さければ、その交点を線分要素の端点候補とし、直線*i*と*j*は接続関係をもつものとする。
4. 1. ~ 3. の処理を全3次元直線に対して行う。端点候補同士の距離が閾値よりも小さい場合、3つの線分要素が1つの端点で接続しているとする。そうでなければ2つの線分要素が1つの端点で接続しているとする。
5. 3つの線分要素*i*, *j*, *k*が接続している端点を物体の頂点と見なす。なお、頂点は3つの線分要素の端点の重心として次式により求める。

$$s_{ijk} = \frac{x_{ij} + x_{jk} + x_{ik}}{3} \quad (18)$$

6. 2つの線分要素が接続している端点  $x_{ij}$  は、その端点を結ぶ直線*i*, *j*のいずれかの両端点に頂点が存在する場合、接続関係を除去して頂点としては扱わない(図10(左))。そうでなければ、頂点と見なす(図10(右))。

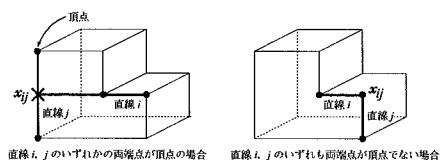


図 10: 2つの線分要素の接続による端点の頂点判定

7. 6. で求めた頂点同士の各距離を算出する。最小距離となる頂点間に接続関係を与え、3つの線分要素で接続した頂点と設定する(図11)。

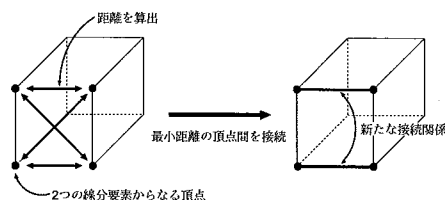


図 11: 新たな接続関係の設定

以上の処理を行うことで、3次元上の特徴線の交点から端点を判定し、線分要素を特定することができる。そして、この線分要素から物体の頂点および線分要素の接続関係を求めることができる。

### 3.3 3次元モデルの生成

#### 3.3.1 線分要素の接続関係による3角面の構成

次に、求めた頂点と線分要素の接続関係を用いて3角面を構成する。ここで、従来手法である文献[3]では、3角面の構成に2次元投影によるドロネー法を拡張して用いている。それは、3次元上の特徴点を円筒面に投影し、投影した平面上で2次元ドロネー法を用いて3角面を構成する手法である。ドロネー法によって構成された特徴点の接続関係を、もとの3次元空間での点群に当てはめることで表面の構成を行っている。しかし、この手法では3次元上の重心からの距離を無視して平面に投影しているため、凹凸の激しい対象物体には適応が難しい。また、円筒面に投影しているため、3次元点群の上下に3角面が構成されないことがあるという問題が生じる。

本研究では、物体の線分要素に基づく頂点や接続関係が求まるので、2次元投影によるドロネー法を適用せず、単純に頂点を結ぶ線分要素の連結関係をもとに3角面を構成する。そのアルゴリズムを以下に述べる。

1. 線分*i*の頂点に対して、接続関係のある線分*j*を選択する。
2. 線分*i*の両端点と線分*j*の*i*とは接続関係のない方の端点を線分で結ぶことで3角面を構成し、その頂点を3角形リストに入れる。



3. 構成した3角面に対し、重複した場合の3角面を除去するために次の処理を行う。
  - (a) 3角面の頂点が3角形リストに存在するか調べる。
  - (b) 3角形リストに同じ頂点が存在する場合、構成した3角面を削除する。
  - (c) 3角形リストに存在しない場合は、構成した3角面の頂点を3角形リストに加える。
4. 線分 $i$ の頂点と接続関係のある全ての線分について1.～3.を行う。
5. 全ての線分要素に対して、1.～4.を行う。

以上の処理により、物体の頂点を結ぶ3角面を構成することができる。しかし、上記の処理により構成した3角面は局所的に重なる面が生じてしまうという問題があるが、本研究ではこの問題について考慮していない。

### 3.3.2 各3角面へのテクスチャ画像選択

次に、生成した3角面にテクスチャを張る。構成された3角面には、その3角面が存在する同一の画像をテクスチャ画像として用いることにより、違和感のないテクスチャマッピングを行う。

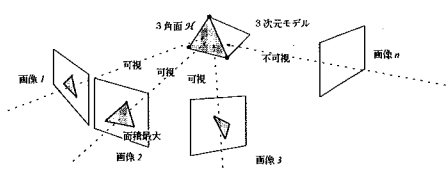


図 12: 各3角面へのテクスチャ画像選択

まず、構成された各3角面に対し、どの画像をテクスチャ画像として利用するか判定する。図 12 に示すように、3角面 $H$ を全画像上に投影し、可視判定を行う。次に可視である画像の中で投影結果が最大面積となる画像を、3角面 $H$ の最適なテクスチャ画像として選択する。

アルゴリズムは次のようになる。

1. 得られた3次元上の各3角面を、全画像上に投影する。

2. 各画像上で可視判定を行い、可視である画像のみを選択する。
3. 2.で得られた画像中より、画像に投影した3角面の面積が、最大となる画像をテクスチャ画像として選択する。

なお、モデルによっては凹部などに3角面が構成されてしまい、どの画像上からも不可視な3角面があることがある。この場合は、3角面を全画像に投影し、面積最大の画像をテクスチャ画像として選択する。

以上のように、画像列から適切な適切なテクスチャ画像を選択することでテクスチャマッピングを行い、3次元モデルを生成する。

## 4 実験と評価

### 4.1 シミュレーション画像列を用いた実験

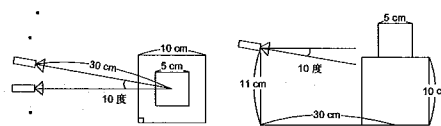


図 13: 物体形状とカメラの位置関係、上面図(左)、側面図(右)

図 13 に示すように物体とカメラを配置した。対象物体は10cmの立方体の上に5cmの立方体が置いてあり、上下の立方体のテクスチャは異なっている。なお、カメラ焦点距離は35mmであり、カメラ焦点位置と物体中心位置までの距離は30cmとする。図 13(左)は、この設定を物体上側から見た図であり、図 13(右)は、物体の側面から見た図である。カメラは、物体を中心として円軌道上を移動しながら、一定間隔で撮影を行う。カメラ位置は下面から11cm上側にあり、物体中心に向けて10度下向である。なお、物体の各面は全て直角している。

以上の環境において、全周囲を10度刻みで36枚撮影する。なお、画像サイズは720×480画素、画像フォーマットはPPM形式である。得られたシミュレーション画像の一部を図 14 に示す。これらの画像列を用いて次の実験を行った。

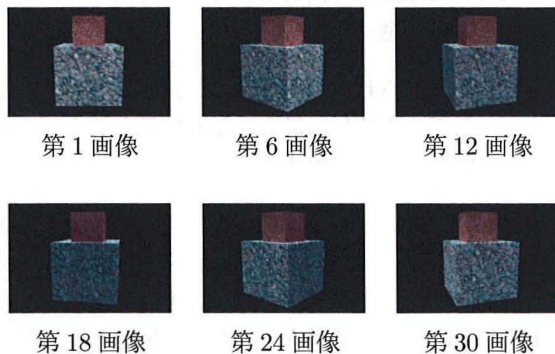


図 14: 実験画像例

#### 4.1.1 特徴点, 特徴線の対応付け検出の評価

まず, 文献 [3] の手法より, 対象画像列にアフィン制約を用いた疎な形状復元を行う. そして, その結果を奥行き制約として利用し, より密な特徴点の対応点探索と追跡を行う. その密に与えられた特徴点の検出結果を図 15 に示す (図中の □ が特徴点を示す).

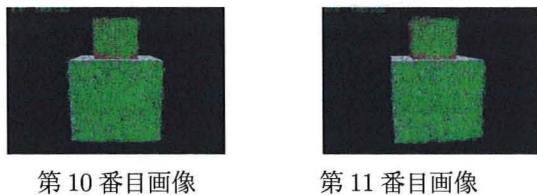


図 15: 奥行き制約を用いて対応付けられた密な特徴点の検出

次に, 本研究で用いる文献 [10] の手法により対象画像列に適用させ, 物体の線分要素とみなしたエッジ上に特徴線の対応付けを行う. 特徴線の対応付けは, 1つの線分上に特徴線を 1本のみ与えており, 全特徴線数は 24 本となる. 対応付けを与えた特徴線の検出結果を図 16 に示す (図中の楕円と線分が特徴線を示す).

奥行き制約を適用して得られた特徴点は, 物体の平面上には多数検出されている. しかし, 線分要素の位置には特徴点の検出が少ないことが確認できる. 一方, 特徴線による対応付けは, 物体の線分要素を表すエッジに与えることができている. 検出したエッジ上の特徴線は, 基準となるエッジの方向に

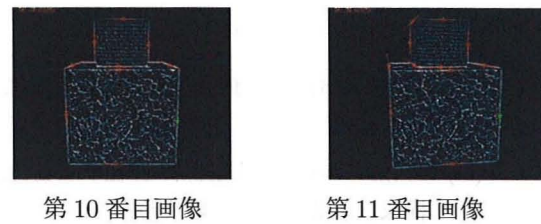


図 16: 線分要素に対応付けた特徴線の検出

対して誤差が生じているものがあるが, 大きく方向が外れているわけではなく, ほぼ安定して対応付けることができている. 3次元モデルを生成する上で, 物体の線分要素は重要な構成要素である. そのため, 特徴点が少ないと, 歪みが生じたモデルが生成されてしまう. したがって, 物体の線分要素には特徴線を用いて対応を与えた方が有効である.

#### 4.1.2 特徴点, 特徴線による形状復元の評価

奥行き制約により得られた密な特徴点の追跡結果を利用して, 非線形最小 2 乗法により復元された特徴点の 3次元座標とカメラ位置・姿勢を図 17 に示す.

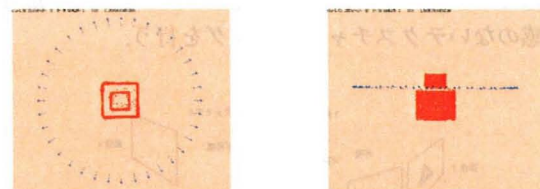


図 17: 奥行き制約を用いた密な特徴点の追跡結果による形状復元, 上図 (左), 横図 (右)

ここで, 中央の密集した点群が 3次元空間中の特徴点を表し, 周囲の円弧状の点がカメラ位置, 点から中央の特徴点群に向かう線がカメラ光軸を表している. なお, 図 17 の 3次元上の特徴点は, 奥行き制約を 2回適用させた結果から, 再投影誤差をもとに信頼性の低い特徴点 1000 点を除去した 11036 点で構成されている. 図 17 を見ると, 物体表面から何点か飛び出している特徴点が見つかる. しかし, カメラ位置は円弧状となっており, 物体の形状はほぼ安定に復元できていることがわかる.

次に、特徴線を用いて非線形最小 2 乗法により復元された特徴線の 3 次元座標とカメラ位置・姿勢を図 18 に示す。ここで、特徴線の 3 次元座標とは、

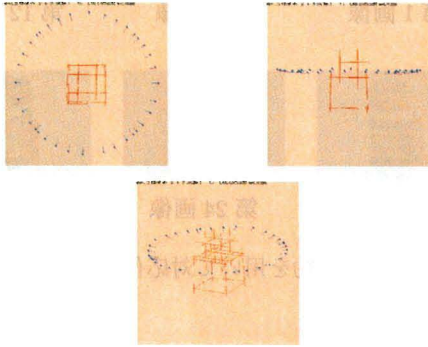


図 18: 特徴線の対応付けによる形状復元, 上図(左), 横図(右), 斜図(下)

エッジ上に特徴線上の 1 点として与えた特徴点の 3 次元座標である。この特徴点を基準として正負に方向ベクトルを与えることで特徴線として表している。なお、中央に密集した線分群が 3 次元空間中の特徴線を表している。周囲の円弧状の点とその点から中央の特徴線群に向かう線は上記と同様、それぞれカメラ位置、カメラ光軸を表している。図 18 を見ると、特徴線の方向が物体の線分要素の方向と比べて誤差を生じてはいるが、カメラ位置は円弧状となっており、ほぼ安定して復元できている。したがって、特徴線を用いることでも物体を安定した形状で復元できることが確認できる。

#### 4.1.3 特徴点、特徴線を用いた形状復元からの 3 次元モデルの生成評価

まず、図 17 の特徴点を用いた形状復元結果に対して、2 次元ドロネー法を適用して 3 角面を構成した結果を図 19 に示す。そして、構成された各 3 角面にテクスチャマッピングを行うことで得られた 3 次元モデルを図 20 に示す。図 20 より、生成された 3 次元モデルは物体の全体形状が復元できており、大きな歪みがないことがわかる。しかし、線分要素に注目すると、歪みが目立って生じていることが確認できる。図 18 の特徴線を用いた形状復元結果に対し、本手法を適用し、求めた物体の頂点と線分要素

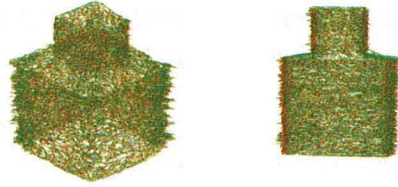


図 19: 2 次元ドロネー法により 3 角面を構成した形状復元, 上図(左), 横図(右)

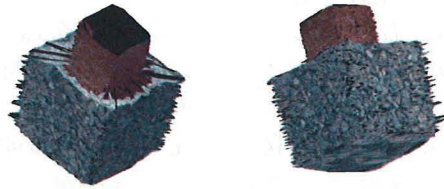


図 20: 文献 [3] の手法により生成した 3 次元モデル, 上図(左), 横図(右)

の連結関係を用いて構成した 3 角面の結果を図 21 に示す。そして、構成された各 3 角面にテクスチャ



図 21: 本手法により頂点間に 3 角面を構成した形状復元, 上図(左), 横図(右)

マッピングを行うことで得られた 3 次元モデルを図 22 に示す。図 22 より、特徴線を用いた形状復元に



図 22: 本手法により生成した 3 次元モデル, 上図(左), 横図(右)



対し、本手法を用いて生成した3次元モデルは、物体の形状がほぼ安定した結果が得られることがわかる。ここで、文献 [3] の手法により生成した3次元モデルと比較してみる。物体の線分要素についての生成結果は、本手法を用いて生成する方が精度が高い結果が得られることが確認できる。

以上より、物体の線分要素については特徴点よりも特徴線を用いた形状復元から3次元モデルを生成する方が有効であることが確認できた。しかし、特徴線を用いた形状復元に基づいて3次元モデルを生成する場合、画像のエッジ検出により生成結果の精度が左右されてしまう。また、物体表面上に凹凸形状が存在していたとしても、構成する3角面が大きいため3次元モデルの表面は平坦に生成されてしまい、違和感の目立つ3次元モデルとなる可能性がある。そのため、エッジ検出法の検討が必要である。また、特徴点と特徴線を統合した形状復元を行うことで、面と線の要素がより高精度な3次元モデルの生成が期待できる。

## 4.2 実画像列を用いた実験

シミュレーション画像列により、本手法による3次元モデルの生成の有効性が確認できた。次に、実画像列を用いて評価実験を行った。

撮影は、対象物体の全周囲をデジタルカメラ (Canon EOS D30) を用いて行った。このとき、焦点距離は一定である。なお、比較対象である保田らの手法では256階調のカラー画像列を用いている。そして、本手法では各画像からエッジ検出を行った濃淡画像列を用いている。

### 4.2.1 「レゴブロック」例

#### 特徴点による形状復元からの3次元モデル生成

レゴブロックの全周囲を約10度刻みで撮影し、36枚の画像列を得た。なお、画像サイズは720×480画素、画像フォーマットはPPM形式である。また、画像の背景は全て白としている。得られた画像列の一部を図23に示す。そして、文献 [3] の手法により奥行き制約を用いて密な対応付けが行われた特徴点の検出結果を図23に合わせて示す (図中の口が特徴点を示す)。

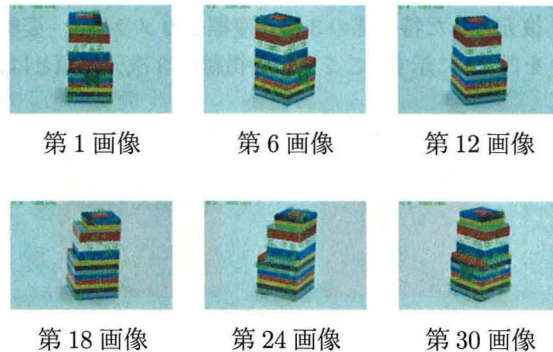


図 23: 奥行き制約を用いて対応付けられた密な特徴点の検出

奥行き制約を適用して得られた特徴点は、物体の平面上に集中して検出されているが、線分要素にはわずか数点しか検出されていない。これは4.1.1節の結果とほぼ同様である。奥行き制約を用いて密な対応特徴点の追跡を2回を行い、ロバスト推定により、信頼性の低い特徴点を300点除去した。その結果、形状を構成する特徴点として1600点が得られた。なお、特徴点の追跡結果のうち3画像間以上で観測された特徴点のみを利用している。ここで、追跡された特徴点の3次元座標とカメラ位置・姿勢の推定結果が得られる。次に、推定した3次元上の特徴点群に対し、2次元ドロネー法により3角面を構成した結果を図24に示す。最後に、図24の3角面で構成された多面体に対して、テクスチャマッピングを行うことで生成した3次元モデルを図25に示す。



図 24: 2次元ドロネー法により3角面を構成した形状復元, 上図(左), 横図(右)

図25を見ると、この手法により実画像列を用いて生成された3次元モデルは、物体全体の大まかな形状は生成することができている。しかし、物体の



図 25: 文献 [3] の手法により生成した 3 次元モデル, 上図 (左), 横図 (右)

線分要素に注目すると, 歪みが頻繁に生じていることがわかる. これは, 図 24 の 3 角面を構成した形状復元結果を見ても, 同部分の復元結果が好ましくないことが確認できる.

### 特徴線による形状復元からの 3 次元モデル生成

次に, 文献 [10] の手法によりエッジ検出を行った画像列の一部を図 26 に示す. また, 物体の線分要素に対応付けた特徴線の検出結果を図 26 に合わせて示す (図中の楕円と線分が特徴線を示す).

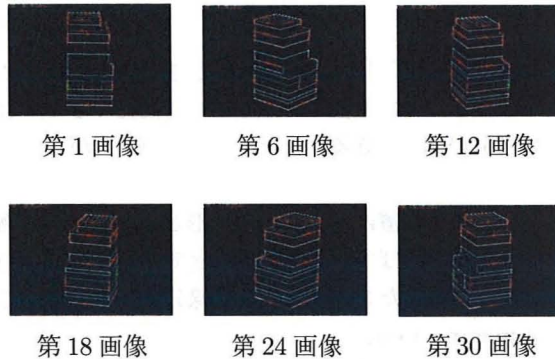


図 26: 線分要素に対応付けた特徴線を検出した画像列の一部

なお, 誤った特徴線の交点を頂点として設定しないようにするため, 対象物体の各立方体を 3 つに分割して特徴線の対応付けを行うものとする. したがって, 物体の線分要素とみなしたエッジに対し, 特徴線を 1 本ずつ対応付けているが, 立方体が重なり線分が共有されているエッジ部分においては, 特徴線の対応付けが 2 本となっている. 対応付けを与えた全特徴線数は 34 本である.

図 26 より, 線分要素に対応付けた特徴線を用いて物体の形状復元を行う. 形状復元された結果を図 27 に示す. 図 27 において, 特徴線を用いた形状復

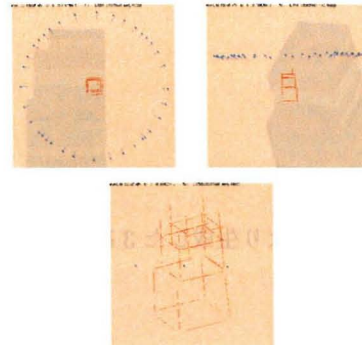


図 27: 特徴線の対応付けによる形状復元, 上図 (左), 横図 (右), 斜図の拡大 (下)

元は, 一部で復元されていない特徴線があるものの, 物体の形状およびカメラ位置・姿勢はほぼ安定している. 復元できなかった特徴線については, 画像列から線分要素としてのエッジが検出されていない, またはエッジ強度が線分要素よりもその付近の方が高く, 推定した特徴線の方向ベクトルが妥当でなかったためである.

次に, 本手法により推定された特徴線の 3 次元上の交点を算出する. 算出された交点から線分要素としての端点を判定し, 物体の頂点と線分要素の連結関係を求める. 線分要素の連結関係に基づいて, 頂点を結び 3 角面を構成した結果を図 28 に示す. 図



図 28: 本手法により頂点間に 3 角面を構成した形状復元, 上図 (左), 横図 (右)

24 と比較して, 物体を構成する線分要素において突出した 3 角面が現れていないことが確認できる.



最後に、図 28 で構成された各 3 角面に対して、撮影画像列からテクスチャマッピングを行い、生成した 3 次元モデルの結果を図 29 に示す。図 29 よ



図 29: 本手法により生成した 3 次元モデル，上図(左)，横図(右)

り、生成された 3 次元モデルは、物体の線分要素において歪みがなく安定した結果となっている。しかし、厳密に観察すると、物体の細かい凹凸部分がすべて平坦となって生成されている。例えば、図 29 の上面において、レゴの結合部分の凹凸がすべて同一平面として生成されている。また、図 29(右)のように真横から物体を観察すると、物体上面部が平坦となっていることが確認できる。これは、4.1.3 節で考察したように、頂点を結ぶ大きい 3 角面を構成するため、その 3 角面内は同一平面上と判定されてしまうことが原因である。したがって、特徴線を用いた形状復元から本手法による 3 次元モデルの生成は、物体平面に凹凸がある場合には違和感が目立ち、形状モデルの精度が低くなってしまう。

#### 4.2.2 「椅子」例

屋外環境下において、広島市立大学大学院別棟の付近にある  $70 \times 120 \times 150$  cm の「椅子」の全周囲を撮影し、36 枚の画像列を得た。なお、画像サイズは  $720 \times 480$  画素、画像フォーマットは PPM 形式である。また、復元物体のみのエッジを検出するため、画像列の背景を全て手動で除去し、黒としている。得られた画像列の一部を図 30 に示す。

図 30 の画像列に示す物体の各線分要素に 1 本ずつ、計 28 本の特徴線を対応付け、復元された特徴線の位置と方向ベクトル、およびカメラ位置・姿勢を図 31(左)に示す。また、本手法を用いて求めた物体の頂点を結び、3 角面を構成した結果を図 31(右)に示す。

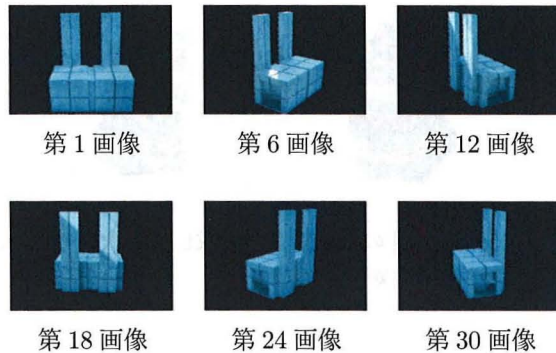


図 30: 実験画像列

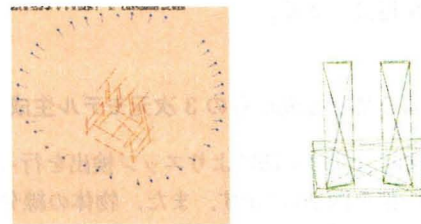


図 31: 特徴線の対応付けによる形状復元(左)，頂点を結び構成した 3 角面(右)

図 31(左)では、特徴線の対応付けを与えることができなかった位置があったため、復元できていない部分が確認できる。また、特徴線の方向が不安定な位置が数本存在していることも確認できる。しかし、カメラ位置は円弧状に近い形となっており、物体の形状をほぼ安定して復元できている。図 31(右)では、構成した 3 角面が物体の線分要素を安定して表現できている。

最終的に、本手法を適用して生成された 3 次元モデルを図 32 に示す。



図 32: 画像列からの 3 次元モデル生成，上図(左)，横図(右)

生成された 3 次元モデルは、線分要素の精度がほ

ば安定していることがわかる。しかし、物体中央の凹凸部分に関しては、構成する3角面により平坦に生成されている。また、テクスチャマッピングによるモデルの表現に違和感が生じている。テクスチャマッピングによる違和感については、局所的に重複する3角面を除去することで解消できると考えられる。

## 5 まとめ

本研究では、直線的な構造からなる人工物を対象とし、その全周囲を撮影した画像列から特徴線を用いた形状復元を行った。その形状復元された直線状の特徴線から線分要素を求め、線分要素が表現が高精度な3次元形状モデルを生成する手法を示した。

まず、本研究では手動で特徴線の位置を与えることを前提とした。文献[10]の手法により、物体形状を構成する輪郭線に対して特徴線による対応付けを行い、その最適な3次元上の位置と方向ベクトルおよびカメラ位置・姿勢を推定した。推定された特徴線の位置と方向ベクトルから物体の3次元形状を復元した。次に、3次元形状を構成している特徴線から3次元上の交点を計算し、端点を判定することで線分要素を特定し、物体の頂点とその頂点を結ぶ線分要素の接続関係を求めた。そして、線分要素の接続関係に基づいて頂点間を結ぶ3角面を構成した。最後に、得られた3角面に対してテクスチャマッピングを行い、線分要素の歪みが軽減した3次元モデルを生成した。

実画像列を用いた実験により、物体の線分要素については、特徴点による対応を追跡するよりも、特徴線による対応付けの方が有効であることが確認できた。さらに、特徴線による3次元形状はほぼ安定した形状を得ることが可能であることを示した。

本手法により生成された3次元モデルは、物体の線分要素に生じる歪みが少なく、従来手法である文献[3]を用いて生成した3次元モデルよりも、視覚的に精度が向上していることが確認できた。しかし、頂点を結ぶことで構成した3角面が大きいため、平面上の細かい形状はすべて同一平面上とみなしてしまう。そのため、物体形状の整合性がとれず、違和感のある3次元モデルが生成される可能性が高い。また、線分要素に特徴線の対応付けを与える段階で、特徴線の位置と方向ベクトルの推定は、す

べて画像中のエッジ強度に依存している。したがって、エッジ強度が低い場合、特徴線の対応付けが困難となり、形状復元が不安定になってしまう。

今後の課題として、特徴線の対応付けが適切に行えるようにするため、物体の直線状な輪郭線におけるエッジ検出精度を高める必要がある。また、今回は特徴線の対応付けに関しては手動で行ったので、特徴線を自動で対応付けることで形状復元を行うことが挙げられる。最後に、生成する3次元モデルの精度をより向上させるために、特徴点と特徴線を統合して形状復元を行うことが考えられる。つまり、平面や複雑に入り組んだ構造をした形状に対しては特徴点、そして線分要素に対しては特徴線を同時に扱うことで、各特徴による形状復元の短所を補強することができることが期待できる。

## 参考文献

- [1] T. Oishi and T. Masuda and K. Ikeuchi: "Digital Restoration of the Cultural Heritages," *Proc. VSMM* pp.934-941, 2002.
- [2] 中川雅朗, 斎藤英雄, 小沢慎治: "ハンディカメラを用いて撮影された画像列からのEPIの補正に基づく室内環境3次元再構築," *信学誌*, Vol.J84-D-II, pp.266-275, 2002.
- [3] 保田和隆, 椋木雅之, 浅田尚紀: "大域的幾何学制約を用いた特徴点追跡に基づく画像列からの3次元モデル生成", *画像の認識・理解シンポジウム (MIRU2005)*, OS5B-19, pp.159-166, 2005.
- [4] C. Tomashi and T. Kanade: "Detection and Tracking of Point Features," *CMU Tech. Rep. CMU-CS-91-132*, Apr. 1991.
- [5] Z. Zhang, R. Deriche, O. Faugeras and Q.-T. Luong: "A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry," *Proc. 3rd Artif. Intell.*, **78**, pp.87-119, 1995.
- [6] 金澤靖, 金谷健一: "段階的マッチングによる画像モザイク生成", *信学論 D-II*, J86-D-II, pp.816-824, 2003.

- [7] M. Okutomi and T. Kanade: "A multiple - baseline stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.15, no.4, pp.353-363, 1993.
- [8] L. Matthies, T. Kanade and R. Szeliski: "Kalman filter-based algorithms for estimating depth from image sequences," *International Journal of Computer Vision*, vol.3, pp.209-239, 1989.
- [9] G. L. Gimel'farb and R. M. Haralick: "Terrain reconstruction from multiple views," *Proc. 7th Int. Conf. on Computer Analysis of Images and Patterns (CAIP'97)*, Kiel, Germany, Sept. 1997. (G. Sommer, K. Daniilidis, and J. Pauli, Eds.). Lecture Notes in Computer Science 1296, Springer: Berlin e.a., pp.694-701, 1997.
- [10] 黒澤 明: "非線形最小 2 乗法を用いた線特徴からの 3 次元形状復元", 広島市立大学大学院情報科学研究科修士論文, (2002).
- [11] 金澤靖, 金谷健一: "コンピュータビジョンのための画像の特徴点抽出", 信学誌, Vol.87, No.12, pp.1043-1048, 2004.
- [12] A. Amano, T. Migita and N. Asada: "Stable Recovery of Shape and Motion from Partially Tracked Feature Points with Fast Non-linear Optimization," *Proc on Vision Interface*, pp.244-251, 2002.
- [13] C. Harris and M. Stephens: "A Combined Corner and Edge Detector," *Proc. Alvey Vision Conf.*, pp.147-151, 1988.
- [14] 田村秀行: "コンピュータ画像処理", 株式会社オーム社, (2002).



# 陰影情報と特徴点对応の統合による 画像列からの3次元形状復元

## 1 はじめに

実物体の3次元モデル化技術はインターネットでのリアルな仮想空間の構築や、デジタルアーカイブにより保存された芸術作品や文化財のウェブ上での公開など多様な目的に用いられる。3次元モデルを生成する手法としては、レーザレンジセンサのような専用の計測装置を用いて得られた情報から3次元モデルを生成する手法がある。しかし、この手法では計測に特殊な装置が必要なことや対象物体がいくつかの撮影条件を満たさなくてはならないという問題がある。そのため、迅速かつ手軽に形状を生成できるシステムが渴望されており、コンピュータビジョン技術の応用が試みられている。

現在までに、様々な方法で画像から3次元形状を復元する手法が提案されている [1]。物体表面での拡散反射は、物体法線と光源との成す角に依存するため、画像から得られる物体の陰影情報 (拡散反射成分) から形状復元が行える。文献 [2] では同一視点の光源状況の違う4枚の画像から法線を推定し、それをもとに反射パラメータや形状を復元し、3次元モデルを生成している。他の手法として物体を異なる位置から撮影した視差のある画像から物体上の特徴点对応に基づき3次元形状を復元する手法もある。文献 [3] では、大局的幾何制約により信頼性の高い特徴点を対応付けし、反復処理を行うことによって3次元モデルを生成している。文献 [4] では屋外環境で動画像数百フレームを入力とし、特徴点の追跡によりカメラパラメータを推定し、マルチベースラインステレオ法を用い3次元モデルを復元している。文献 [5] では物体の姿勢や視点位置の変化を伴う画像列から物体の3次元形状を復元し、対象物体が凸物体であり、表面反射が Lambert モデルに従うとして光学的特性を推定している。

しかし陰影情報を用いる手法 [2] では、形状に関して、法線が連続していると仮定しているため法線が連続していない物体や、隣り合う画素が違う面であり3次元位置が急激に変化する場合適切な形状が得られないという問題がある。また形状復元の際、

法線の推定誤差が近傍画素に影響し適切な復元結果とならない場合がある。一方特徴点对応を用いた手法 [3][4][5] では特徴点が画像の画素数にくらべて少数であるので3次元モデルを生成する際に細かな形状を表現できない場合がある。

本研究では、対象物の画素毎の3次元座標を推定し高精細な3次元形状を復元することを目的とする。そのため同じ視点で異なる光源状況下の4枚の画像から得られる画素毎の法線情報と、特徴点对応を用いた3次元モデル生成手法から得られる特徴点の3次元座標を統合する。まず物体は法線が連続する面に分割できると仮定し、法線情報を用いたラベリングにより各面に分割する。次に各面における特徴点の3次元座標をもとに、法線情報を用いて奥行の推定を行う。これにより不連続な面を含む対象物の各画素に対する3次元座標を推定する。

本論文では、まず、2節で陰影情報を用いる従来手法と特徴点を用いる従来手法を示す。次に3節では、本手法で用いる陰影情報から法線情報を推定する手法と特徴点から3次元座標を推定する手法の原理を述べ、ラベリングと最適化の手法について述べる。そして、4節では本手法を適用した実験結果を示す。最後に5節でまとめとする。

## 2 画像列からの3次元モデル生成

### 2.1 陰影情報を用いた3次元モデル生成

陰影情報を用いた3次元モデル生成の手法として文献 [2] がある。文献 [2] では、まず図1に示すようにカメラ位置・姿勢を固定し、位置が既知である4つの光源状況下において、1画像中に1つの光源を利用し、合計4枚の画像を撮影する。次に、それらの画像を用いて表面反射特性 (拡散反射係数および鏡面反射係数) と表面法線を推定することで3次元モデルを生成している。ここで、対象物体は滑らかである (法線が連続している) とする。また物体上に RGB 値の差 (色度) が大きい画素が存在し

ているものとする。以下に処理のアルゴリズムを述べる。

### 1. 色度分析による表面法線の推定

画像上で、RGB 値間の差 (色度) が大きな画素に対して光源方向ベクトルと色度から表面法線を求める。

### 2. 鏡面反射特性の推定

推定された表面法線をもとに、拡散反射成分の線形特性を利用して観測された画像上で拡散反射成分を除去する。残った成分が鏡面反射成分となるため、その特性 (鏡面反射係数) を推定する。

### 3. 拡散反射特性の推定

表面法線が推定された各点に対して拡散反射成分からその特性 (拡散反射係数) を推定する。

### 4. 表面法線の再推定

色度分析で法線が得られなかった点についてまず、画素値が拡散反射成分のみであると仮定して拡散反射モデルに基づき、表面法線を推定する。次に、表面法線が連続しているか検証する。連続していない画素については、画素値において鏡面反射成分が優位であるとみなし、鏡面反射モデルに基づき表面法線を再推定する。

### 5. 表面モデルの推定

2~4 の反復計算を行い、表面法線を推定する。

### 6. 表面形状の復元

再推定された表面法線をもとに文献 [7] の手法を用いて表面形状を復元する。

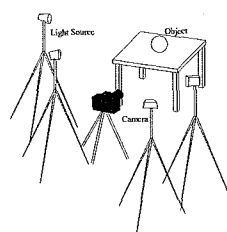


図 1: 各光源の切り替えによる画像の撮影

## 2.2 特徴点对応を用いた 3 次元モデル生成

画像列から特徴点对応を用いて形状復元し、テクスチャを付加した 3 次元モデルを生成する手法として文献 [3] がある。文献 [3] ではまず、図 2 に示すように、対象物を中心にして、カメラが 1 方向に移動 (円運動) しながら対象物の全周囲を撮影する。次に、その全周囲画像列を用いて対象物体の全体構造に基づく大局的な幾何制約により特徴点探索することで 3 次元点群により表現された形状復元を行う。そして、その推定結果に対してテクスチャ画像を付加した 3 次元モデルを自動生成する。処理手順を図 3 および以下に示す。

### 1. 特徴点の検出

対象を撮影した画像列を入力とし、各画像から Harris の特徴点検出手法 [6] を用いて、特徴点を数百点検出する。

### 2. 特徴点の対応付けと追跡

各画像から検出された特徴点に対して、画像全体をアフィン変換で近似するという大局的幾何制約を用いて、対応付けと追跡を行う。

### 3. ロバスト推定を用いた形状復元とカメラ位置・姿勢の推定

特徴点の追跡結果に対して、仮の形状復元を行う。仮の復元結果を用いて各特徴点の信頼性を評価し、信頼性の低い特徴点を除去する。次に、残った確からしい特徴点を元にしてカメラ位置・姿勢および特徴点の 3 次元座標を再度推定する。

### 4. 2 次元投影によるドロネー法を用いた 3 角面生成

3 次元上の特徴点座標を、円筒面に投影した点群に対してドロネー法を適用し、3 角面を生成する。

### 5. 3 次元形状の大局的幾何制約を用いた密な対応点探索と追跡

特徴点で構成された 3 角面と、カメラ位置・姿勢を用いることで、エピポーラ線上の対応点探索範囲を限定し、より密な特徴点の対応点探索と追跡を行う。3. ~ 5. までは繰り返して行ない、追跡される特徴点数が収束するま

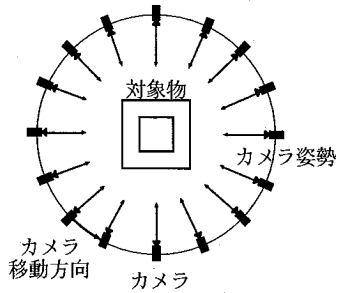


図 2: カメラの円運動による全周囲画像列の撮影

で繰り返す。最後に、特徴点の再追跡を行うことで、各特徴点の追跡画像数を増やし、同一の特徴点を連結する。

### 6. テクスチャ画像選択

得られた3角面の3次元座標と隣接関係を用いて、全周囲画像列より各3角面に適切なテクスチャ画像を選択し、3次元モデルを生成する。

以上のように、画像情報に基づく大局的制約を用いた形状復元、3次元形状の大局的制約を用いたモデル生成を段階的に適用する。この際、特徴点の追跡結果は必然的に誤差が含まれてしまうものと考え、前段階の処理の誤差を許容しながら段階的に誤差を減らすことで適切な3次元モデルの生成を行う。

## 2.3 従来手法の問題点と本手法での解決法

上記の2つの手法により、画像列から3次元モデルを生成できる。しかし、これら2つの手法には以下に述べる問題点がある。

陰影情報を用いた3次元モデルの生成手法では物体表面の法線が連続していると仮定しており、法線が連続していない場合は適切な3次元形状を復元できない。また3次元形状の推定は法線情報のみを用いているため、推定誤差が近傍画素に影響し、不適切な復元結果となる場合がある。

また、特徴点対応を用いた3次元モデルの生成手法においては特徴点を頂点とする3角面を生成し、その3角面に対してテクスチャ画像を選択している

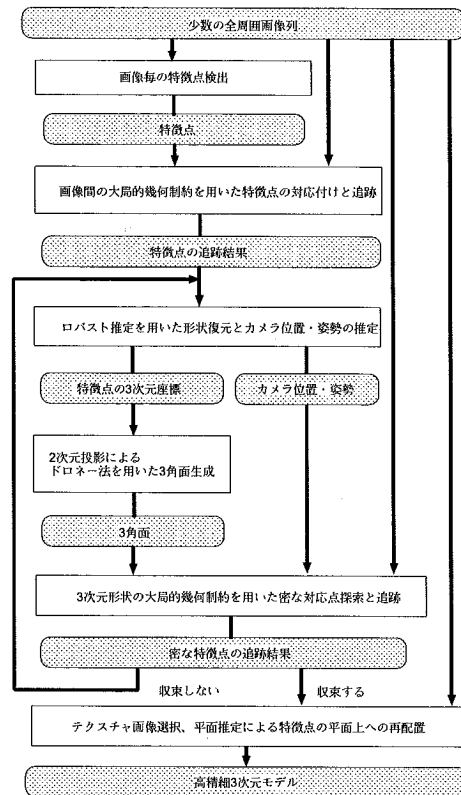


図 3: 文献 [3] の処理の流れと入力・出力データの関係

が、特徴点は画素数に比べて少数であるため細かな形状を表現できない。

そこで本手法では上記の問題点を解決することを目的とする。そのために、陰影情報から得られる法線と特徴点対応により得られる特徴点の3次元座標を統合する。この際、法線方向が連続した面への分割を行い、各面で法線と特徴点の3次元座標を用いて形状復元を行う。これにより、陰影情報からは画素毎の奥行情報、特徴点対応から特徴点の3次元座標と面の位置関係を得ることで、高精細な形状を復元する手法を提案する。

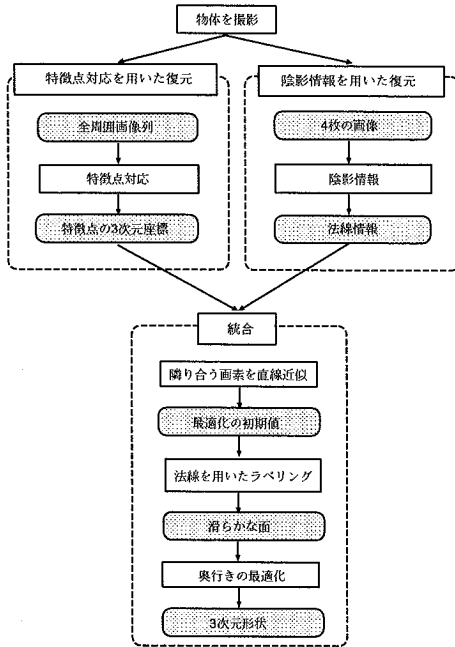


図 4: 本手法の処理手順

### 3 陰影情報と特徴点対応を利用した 3次元形状復元手法

#### 3.1 処理手順

本手法の処理手順を以下の図 4 に示す。また、その処理アルゴリズムを以下に述べる。

##### 1. 陰影情報からの法線方向の算出

同一視点位置において異なる光源状況下の 4 枚の画像を撮影し、陰影情報から法線情報を得る。ここで物体色は色度が大きいと仮定する。

##### 2. 特徴点の 3次元座標の算出

文献 [3] の手法により、全周囲画像列から特徴点の 3次元座標を得る。このうち 1つの視点は 1. の視点と同一とする。

##### 3. 3次元形状の復元

推定された法線同士の成す角が小さいものでラベリングを行い、各面に分割する。そして 1. と 2. で得られた情報をもとに、ラベル別に奥行き最適化を行うことで物体上の画素毎の 3次元座標を推定し、形状復元を行う。

#### 3.2 陰影情報からの法線方向の算出

文献 [2] の手法をもとに陰影情報から画素毎の法線情報を得る。まず、同じ視点から異なる光源状況下で撮影した画像を用いる。ここで光源は位置が既知である 4つの光源を用いるが、点灯する光源は画像中に 1個とし、遠方から照らしているとする。また全ての光源は同じ明るさの白色光であるとする。

一般に、物体の表面反射特性は拡散反射成分と鏡面反射成分から構成される。拡散反射成分は物体表面色に直接寄与し、物体表面の法線と入射光ベクトルによって変化する。一方、鏡面反射成分の色度は入射光とほぼ一致する。拡散反射成分が Lambert のモデル [9](図 5) に従い、鏡面反射は Phong のモデル [10](図 6) に従うとすると、ある地点  $\mathbf{x}$  における反射光は次式で表せる [11]。

$$\begin{aligned} I(\mathbf{x}) &= I_{diff}(\mathbf{x}) + I_{spec}(\mathbf{x}) \\ &= L_i K_d \cos\theta_d + L_i K_s \cos^\gamma\theta_s \end{aligned} \quad (1)$$

ここで  $I_{diff}(\mathbf{x})$  は拡散反射成分、 $I_{spec}(\mathbf{x})$  は鏡面反射成分である。拡散反射成分について、 $L_i$  は光源の持つ輝度値、 $K_d = \{\rho_r, \rho_g, \rho_b\}$  は物体表面における拡散反射係数を示している。また、 $\theta_d$  は図 5 のように物体表面における法線  $\bar{\mathbf{N}}$  と光源方向ベクトル  $\bar{\mathbf{L}}$  とのなす角であり、 $\cos\theta_d = (\bar{\mathbf{N}} \cdot \bar{\mathbf{L}})$  に書き換えることができる。また鏡面反射成分について、 $K_s$  は鏡面反射係数、 $\gamma$  はハイライトの広がりを表す鏡面反射指数である。また、 $\theta_s$  は図 6 のように光源方向ベクトル  $\bar{\mathbf{L}}$  における正反射ベクトル  $\bar{\mathbf{R}}$  と視線方向ベクトル  $\bar{\mathbf{V}}$  とのなす角であり、 $\theta_s$  が  $0^\circ$  のとき最大の鏡面反射が生じ、 $\theta_s$  が増加するとともに鏡面反射成分は急激に減衰している。この減衰は鏡面反射指数  $\gamma$  によって変化する。一般に、 $\gamma$  は 1 以上の実数として定義される。

反射光が拡散反射成分と鏡面反射成分に分けられる場合、画素の色度と光源の色度が一致しない場合は色度情報から鏡面反射成分を除去することが可能であり、その画素に対応する物体表面の幾何情報が得られる。光源が白色の場合は式 (1) より以下が成り立つ。

$$\begin{aligned} \hat{i}^k &= i_{\max}^k(\mathbf{x}) - i_{\min}^k(\mathbf{x}) = \xi_o^k b(\mathbf{x}) \cos\theta_d \quad (2) \\ b(\mathbf{x}) &= \rho_{\max}(\mathbf{x}) - \rho_{\min}(\mathbf{x}) \end{aligned}$$

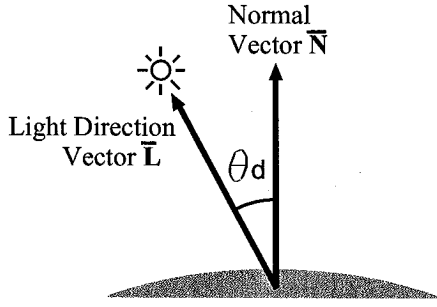


図 5: Lambert モデル

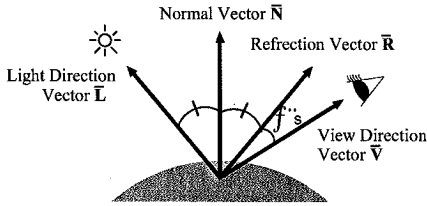


図 6: Phong モデル

ここで、 $i_{\max}$  は色ベクトル  $I = \{i_r, i_g, i_b\}$  の最大色であり、 $i_{\min}$  は最小の色である。また  $k$  は  $k$  番目の光源を表す。  $\xi_o$  は光源色  $L_i = \{\xi_r, \xi_g, \xi_b\}$  で光源色が白色の場合、 $\xi_o = \xi_r = \xi_g = \xi_b$  である。  $\cos\theta_d$  は  $(\bar{\mathbf{N}} \cdot \bar{\mathbf{L}})$  に書き換えることができるため、式 (2) より得られた  $\hat{\mathbf{I}} = (\hat{i}^1 \cdots \hat{i}^k)^T$  を用いて、表面法線は次式で導出可能である。

$$\begin{aligned} b(\mathbf{x})[\tilde{\mathbf{L}}]\bar{\mathbf{N}} &= \hat{\mathbf{I}} \\ b(\mathbf{x})[\tilde{\mathbf{L}}]^T[\tilde{\mathbf{L}}]\bar{\mathbf{N}} &= [\tilde{\mathbf{L}}]^T\hat{\mathbf{I}} \\ b(\mathbf{x})\bar{\mathbf{N}} &= ([\tilde{\mathbf{L}}]^T[\tilde{\mathbf{L}}])^{-1}[\tilde{\mathbf{L}}]^T\hat{\mathbf{I}} \quad (3) \end{aligned}$$

$b(\mathbf{x})$  はスケールファクタである。また  $\bar{\mathbf{L}}^k = \xi_o^k \bar{\mathbf{l}}^k$  であり、 $[\tilde{\mathbf{L}}] = (\bar{\mathbf{L}}^1 \cdots \bar{\mathbf{L}}^k)^T$  は光源マトリクスである。  $\bar{\mathbf{l}}^k$  は  $k$  番目の光源方向を定義する単位ベクトルである。

### 3.3 特徴点の 3 次元座標の算出

#### 3.3.1 カメラモデルと投影モデル

図 7 に示すように、現実空間をモデル化した 3 次元空間を考える。そこに固定された座標系を世界座標系とする。次に、その 3 次元空間を自由に動き回るカメラを考える。ここで、カメラの焦点位置  $C$  を原点とした局所座標系をカメラ座標系とする。

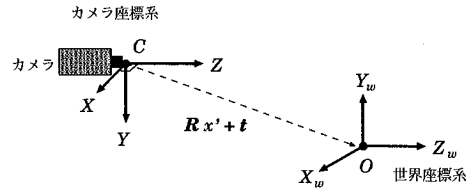


図 7: カメラ座標系と世界座標系との関係

ある点のカメラ座標系における座標  $\mathbf{x} = (x, y, z)^T$  と、世界座標系における座標  $\mathbf{x}' = (x', y', z')^T$  との関係は次のように表される。

$$\mathbf{x} = \mathbf{R}\mathbf{x}' + \mathbf{t} \quad (4)$$

ここで、 $\mathbf{t}$  は並進ベクトル、 $\mathbf{R}$  は回転行列である。これらは世界座標系におけるカメラの位置と姿勢を表すパラメータである。以後簡単のために、これらのパラメータをカメラ位置・姿勢と呼ぶ。

次に、物体と、カメラで撮影された物体の投影される画像平面の関係を図 8 に示す。カメラの焦点位置  $C$  から画像平面  $I$  までの距離  $f$  を焦点距離と呼ぶ。焦点位置  $C$  を通り画像平面  $I$  に垂直な線を光軸、光軸と画像平面  $I$  との交点  $c$  を画像中心と呼ぶ。

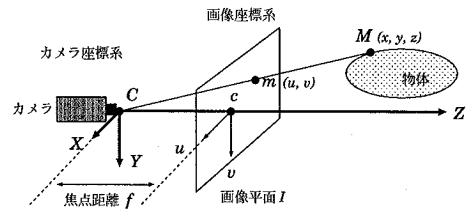


図 8: カメラ座標系と画像座標系との関係

また画像中心  $c$  を原点とし、 $u$  軸と  $v$  軸をそれぞれ画像の画素の配列軸に合わせた 2 次元座標系を

画像座標系と呼ぶ。ここで画像に対して右向きに  $u$  軸、下向きに  $v$  軸をとる。カメラ座標系は、焦点位置  $C$  を原点に光軸を  $Z$  軸とし、 $Z$  軸と直交する  $X$  軸と  $Y$  軸はそれぞれ画像座標系の  $u$  軸と  $v$  軸に一致する。

3次元空間中の点  $M$  が、画像平面  $I$  上で  $m$  に投影されたとすると、点  $M$  とカメラの焦点位置  $C$ 、画像平面  $I$  上の像  $m$  は一直線上にある。この関係を透視投影 (Perspective Projection) という。前述の座標系を用いると、3次元空間内の点  $M$  のカメラ座標系での座標  $(x, y, z)^T$  とその2次元像  $m$  の画像座標  $(u, v)^T$  との間に次式のような関係が成り立つ。

$$\begin{bmatrix} u \\ v \end{bmatrix} := \mathcal{P} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (5)$$

$$\text{ここで } \mathcal{P} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \frac{f}{z} \begin{bmatrix} x \\ y \end{bmatrix} \quad (6)$$

ここで  $\mathcal{P}$  は透視投影を表す演算子である。

点  $M$  を世界座標系での座標  $(x', y', z')^T$  で表わすと、2次元像  $m$  は世界座標系からカメラ座標系への変換を含む次式で表される。

$$\begin{bmatrix} u \\ v \end{bmatrix} = \mathcal{P} \left[ \mathbf{R} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} + \mathbf{t} \right] \quad (7)$$

ここで、 $\mathbf{R}$ ,  $\mathbf{t}$  はカメラ位置・姿勢である。

### 3.3.2 特徴点

画像列からの形状復元で利用される点を**特徴点**と呼ぶ。この特徴点は3次元空間中の点であり、投影された特徴点は画像列上で観測される。画像から対応の取りやすい点を特徴点として選ぶことを**特徴点検出**という。形状復元を行うためには、異なる画像上に投影された同一の3次元上の特徴点を対応付ける必要がある。この特徴点の対応を決めることを**対応点探索**という。対応点探索を用いて、2画像間の特徴点を連結することを**特徴点の対応付け**といい、画像列中の対応する特徴点同士を全て連結する処理を**特徴点の追跡**という。なお、特徴点の追跡により得られたデータは**特徴点の追跡結果**と呼ぶことにする。

### 3.3.3 非線形最小2乗法による特徴点3次元座標の推定

多視点画像での特徴点の追跡結果をもとに物体の形状と各画像を撮影したカメラの位置・姿勢を同時に推定する。この問題は、再投影誤差の最小化として定式化でき、非線形最適化計算により解くことができる。

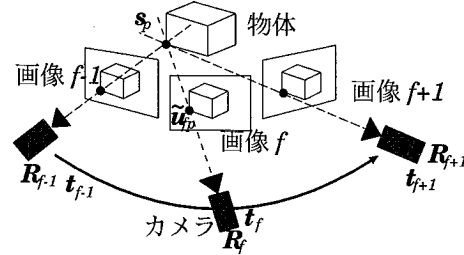


図 9: 形状復元とカメラパラメータ推定

図 9 に示すように、第  $f$  ( $1 \leq f \leq F$ ) 画像で観測された第  $p$  ( $1 \leq p \leq P$ ) 特徴点の観測座標を  $\tilde{u}_{fp} = (\tilde{u}_{fp}, \tilde{v}_{fp})$ 、第  $p$  特徴点の3次元座標を  $s_p = (x_p, y_p, z_p)$ 、第  $f$  画像のカメラ位置・姿勢を  $\mathbf{R}_f$  (回転行列)、 $\mathbf{t}_f$  (並進ベクトル) とする。  $s_p$  を  $\mathbf{R}_f$  と  $\mathbf{t}_f$  のカメラで投影した時の画像座標を  $u_{fp}$  とすると、画像列からの形状復元は次のように定式化される。

$$\arg \min_{s_p, \mathbf{R}_f, \mathbf{t}_f} \sum_{(f,p)} |\tilde{u}_{fp} - u_{fp}|^2$$

$$\text{ここで } u_{fp} = \mathcal{P}(\mathbf{R}_f s_p + \mathbf{t}_f) \quad (8)$$

ここで、 $\mathcal{P}$  は透視投影変換を表す。式 (8) より  $s_p$ ,  $\mathbf{R}_f$ ,  $\mathbf{t}_f$  を求める。

この式は、図 10 に示すように特徴点の3次元座標  $s_p$  をカメラ位置・姿勢  $\mathbf{R}_f$ ,  $\mathbf{t}_f$  で投影した場合の投影座標  $u_{fp}$  と、観測座標  $\tilde{u}_{fp}$  との誤差 (以下、再投影誤差と呼ぶ) の2次ノルムの総和が最小になる  $s_p$ ,  $\mathbf{R}_f$ ,  $\mathbf{t}_f$  を推定するものであり、観測座標に等方性正規分布の誤差を仮定した場合の最尤推定を与える式である。この式を最小化することにより、最尤推定の意味で最適な形状復元が行える。

なお、 $s_p$ ,  $\mathbf{R}_f$ ,  $\mathbf{t}_f$  が未知数なので、式 (8) は非線形問題となる。非線形最適化では、繰り返し処理により解を逐次更新する反復法が用いられる。計算

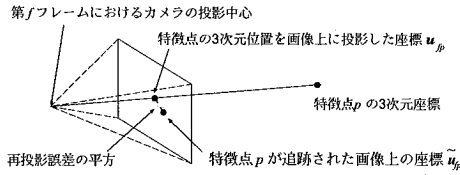


図 10: 再投影誤差

方法としては、Levenberg Marquardt 法などの任意の汎用非線形最適化手法が利用できるが、問題の規模が大きい場合、より計算量の少ない、前処理付き共役勾配法が適している [8].

### 3.4 法線情報と特徴点座標の統合

#### 3.4.1 原理

3.2 節で得られた法線情報と、3.3 節で得られた特徴点に関する 3次元座標をもとに 1画素毎の 3次元座標を求める。この場合、文献 [12] の手法を変形した式 (9) を用いる。法線情報から奥行き  $z$  を求めるエネルギー関数の最小化は次式となる。

$$\begin{aligned}
& \arg \min_{z_{u,v}} E(z_{u,v}) \\
&= \arg \min_{z_{u,v}} \sum_{u,v} \lambda \left( \left( \frac{\partial z}{\partial x} - p_{u,v} \right)^2 + \left( \frac{\partial z}{\partial y} - q_{u,v} \right)^2 \right) \\
&\quad + (1 - \lambda) \left( \left( \frac{\partial^2 z}{\partial x^2} \right)^2 + \left( \frac{\partial^2 z}{\partial y^2} \right)^2 \right) \\
&= \arg \min_{z_{u,v}} \sum_{u,v} \lambda \left( \left( \frac{1}{2} \left( \frac{z_{u+1,v} - z_{u-1,v}}{\Delta x} \right) - p_{u,v} \right)^2 \right. \\
&\quad \left. + \left( \frac{1}{2} \left( \frac{z_{u,v+1} - z_{u,v-1}}{\Delta y} \right) - q_{u,v} \right)^2 \right) \\
&\quad + (1 - \lambda) \left( \left( \frac{z_{u+1,v} - 2z_{u,v} + z_{u-1,v}}{\Delta x} \right)^2 \right. \\
&\quad \left. + \left( \frac{z_{u,v+1} - 2z_{u,v} + z_{u,v-1}}{\Delta y} \right)^2 \right)
\end{aligned}$$

添字の  $u, v$  は各画素に対応している。  $z_{u,v}$  は地点  $u, v$  での奥行きを表し、  $(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, 1)$  は法線である。また、3.2 節で求めた法線を  $(p_{u,v}, q_{u,v}, 1)$  とする。  $(\frac{\partial^2 z}{\partial x^2})^2 + (\frac{\partial^2 z}{\partial y^2})^2$  は滑らかさ制約であり、隣接する点が直線に近いほど 0 に近くなる。  $\lambda$  は重みづけであり、法線情報を用いる場合と滑らかさ制約の内どちらを重視するかを決めるものである。この式は各画素  $(u, v)$  に対応する奥行き  $z_{u,v}$  を未知パラメー

タとして 3.2 節で算出した法線情報と整合するような  $z_{u,v}$  を誤差最小化により求めるものである。この際、滑らかさ制約により、奥行き  $z_{u,v}$  が滑らかに変化するようにしている。又、3.3 節で求めた特徴点については  $z_{u,v}$  の値が既知であるので、その値を用いてその他の点の奥行きを求める。

式 (9) の微分を差分により近似して離散化すると式 (9) が得られる。ここで  $\Delta x$  と  $\Delta y$  はそれぞれ 1画素分の  $x, y$  の変化量である。

#### 3.4.2 共役勾配法

最適化の解法として共役勾配法がある [13]. 本論文ではエネルギー関数の最小化を解くために共役勾配法を用いる。共役勾配法では以下の式を繰り返す。

$$\begin{aligned}
\mathbf{x}_0 &= \text{初期値} \\
\mathbf{d}_0 &= \mathbf{g}_0 \\
\mathbf{x}_{k+1} &= \mathbf{x}_k - \alpha_k \mathbf{d}_k \quad \text{ただし } \alpha_k \text{ は } E(\mathbf{x}_k - \alpha_k \mathbf{d}_k) \text{ を最小化すること} \\
\mathbf{d}_{k+1} &= \mathbf{g}_{k+1} + \beta_k \mathbf{d}_k \quad \text{ただし } \beta_k \text{ は } \mathbf{d}_{k+1} \text{ と } \mathbf{d}_k \text{ を共役とすること}
\end{aligned}$$

ここで  $\mathbf{g}_k = \nabla E |_{\mathbf{x}}$  とする。また  $\mathbf{d}_{k+1}$  と  $\mathbf{d}_k$  を共役とする  $\beta_k$  の選び方としては以下の Fletcher-Reeves の公式 (9) を用いる。

$$\beta_k = \frac{\mathbf{g}_{k+1}^T \mathbf{g}_{k+1}}{\mathbf{g}_k^T \mathbf{g}_k} \quad (9)$$

また共役勾配法の中の  $E(\mathbf{x}_k - \alpha_k \mathbf{d}_k)$  を最小化する  $\alpha_k$  を求めるために次式を考える。

$$\arg \min_{\alpha} E(\mathbf{x} - \alpha \mathbf{d}) \quad (10)$$

これは、現在位置  $\mathbf{x}$  と探索方向  $\mathbf{d}$  が決まったときに  $E(\mathbf{x} - \alpha \mathbf{d})$  を最小にする  $\alpha$ 、即ち  $\mathbf{d}$  の方向に沿った最適な更新量を決めるものである。今回共役勾配法の一部として使うため厳密な最小化は必要なく、多少緩和された条件を満たす  $\alpha$  を満たせば十分である。

目的関数  $E(\mathbf{x} - \alpha \mathbf{d})$  は  $\alpha$  の関数  $f(\alpha)$  と見なせる (図 11). これを 2 次関数  $g(x)$  で近似する。この場合、

等間隔の3点で  $g(x)$  と  $f(\alpha)$  が一致するようにする。これら3点での関数値を  $g_i := g(i) = f(\alpha)$ ,  $i \in 0, 1, 2$  と書く (図 12)。  $s$  は適切なスケールである。最適点  $x^*$  では  $g'(x^*) = 0$  を満たすので、  $x^* = 1 + \delta$  とおくと、

$$\begin{aligned} g'(x^*) &= g'(1 + \delta) = g'(1) + \delta g''(1) = 0 \\ g'(1) &= (g_2 - g_0) / 2 \\ g''(1) &= (g_0 - 2g_1 + g_2) \end{aligned}$$

よって、  $x^*$  は次式となる。

$$x^* = 1 - \frac{g_2 - g_0}{2(g_0 - 2g_1 + g_2)} \quad (11)$$

$f(\alpha)$  が 2 次関数の場合、この方法で最適解  $\alpha^* = sx^*$  が得られるが、数値的安定性を考慮すると  $g_1 < g_0$ ,  $g_1 < g_2$  を満たしていることが望ましい (図 12) ので、以下の操作を繰り返し、  $s$  を適切に選ぶ。

$g_1 > g_0$  の場合 (図 13),  $s \rightarrow s/2$ .

$g_1 > g_2$  の場合 (図 14),  $s \rightarrow 2s$ .

このように  $s$  を選ぶことによって、  $x^*$  の発散を防ぐ効果がある。

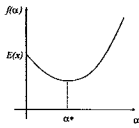


図 11: 1次元最適化

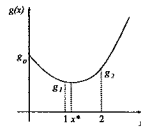


図 12: 最適条件

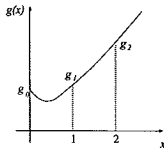


図 13: 縮小

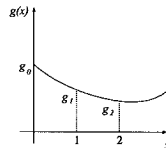


図 14: 拡大

### 3.4.3 最適化

以上の最適化処理から奥行き  $z$  を各画素について求めることによって 3 次元形状を復元する。特徴点における奥行き  $z$  については、その 3 次元座標が信頼できる場合は最適化の更新は行わない。しかし特

徴点の 3 次元座標が信頼できない場合は一定回数更新せずに最適化を行った後、特徴点も含めた最適化処理を行う。

#### 端点の処理

最適化計算により奥行き  $z$  は算出できる。しかし、端点の更新に関しては以下に述べる問題がある。

まず更新に用いる地点  $(a, b)$  の微分  $\frac{\partial E}{\partial z_{a,b}}$  を次式に示す。

$$\begin{aligned} \frac{\partial E}{\partial z_{a,b}} &= \left( -\frac{\lambda}{2} + 2(1 - \lambda) \right) (z_{a+2,b} + z_{a,b+2} + z_{a-2,b} + z_{a,b-2}) \\ &\quad - 8(1 - \lambda) (z_{a+1,b} + z_{a,b+1} + z_{a-1,b} + z_{a,b-1}) \\ &\quad + (2\lambda + 24(1 - \lambda)) z_{a,b} \\ &\quad + p_{a,b} - p_{a,b-1} + q_{a,b} - q_{a-1,b} \end{aligned} \quad (12)$$

式 (12) に示す通り、更新には注目画素とその上下左右 2 画素ずつの奥行情報を用いている。この場合、物体の端では物体上ではない画素の奥行情報を更新に用いている。物体上ではない奥行きは更新していないため、その奥行きが正しくない場合は適切な最適化結果が得られない。この問題を解決するため、推定した法線情報のみを用いて周辺画素の奥行きを更新し、最適化を行う。

その方法を図 15 を用いて説明する。図 15 中の (A), (B) は法線が推定されていない画素であり、(C), (D) は法線が推定された物体の端の 1 画素目、2 画素目である。最適化としてまず、(A), (B) の法線には (C) と同じ法線を与える。

そして、図 15 の (B), (C), (D) の最適化のエネルギー関数として、まず式 (9) において滑らかさ制約をなくす。そして  $(\frac{1}{2}(\frac{z_{u+1,v} - z_{u-1,v}}{\Delta x}) - p_{u,v})^2$  の部分を、図 15(a) のように  $x$  軸の一方の法線が推定されていない場合は式 (13)、図 15(b) のように  $x$  軸方向の + 方向の法線が推定されていない場合は式 (14) にそれぞれ変えたものを用いる。

$$\sum_{u,v} \lambda \left( \frac{(z_{u+1,v} - z_{u,v})}{\Delta x} - p_{u,v} \right)^2 \quad (13)$$

$$\sum_{u,v} \lambda \left( \frac{(z_{u,v} - z_{u-1,v})}{\Delta x} - p_{u,v} \right)^2 \quad (14)$$

同様に、  $y$  軸の一方の法線が推定されていない場合は式 (15)、  $y$  軸方向の + 方向の法線が推定されていない場合は式 (16) に変えたものを用いる。



$$\sum_{u,v} \lambda \left( \frac{(z_{u+1,v} - z_{u,v})}{\Delta y} - q_{u,v} \right) \quad (15)$$

$$\sum_{u,v} \lambda \left( \frac{(z_{u,v} - z_{u,v-1})}{\Delta y} - q_{u,v} \right) \quad (16)$$

また、図 15 の地点 (A) の奥行き  $z$  の与え方としては、地点 (B) の法線に垂直な直線上にあるとしてそれぞれ以下の式で奥行き  $z$  を算出する。

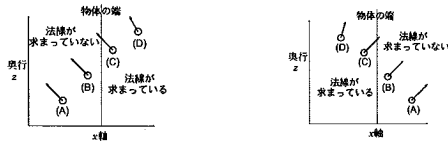
$$z_{u-1,v} = z_{u,v} + p_{u,v} * \Delta x \quad (17)$$

$$z_{u+1,v} = z_{u,v} - p_{u,v} * \Delta x \quad (18)$$

$$z_{u,v-1} = z_{u,v} + q_{u,v} * \Delta y \quad (19)$$

$$z_{u,v+1} = z_{u,v} - q_{u,v} * \Delta y \quad (20)$$

これにより、物体の端の内側 1 画素目、2 画素目と外側 1 画素目では更新に注目画素と前後 1 画素を使い、物体の端の外側 2 画素目は外側 1 画素目の直線上にあるので、最適化は物体の法線情報のみを用いた推定が行われる。



(a)  $u$  軸の一方向の法線が推定されていない (b)  $v$  軸の+方向の法線が推定されていない

図 15: 物体周辺の最適化の方法

### 3.4.4 ラベリング

3.4.1 節で用いた最適化の式では、隣接する画素同士が滑らかに繋がっていると仮定している。この場合、隣接する画素同士が違う面であり、3次元位置が急激に変化している位置では適切な最適化結果が得られず、片方の面から滑らかに繋がる最適化結果となる。このことを解決するために法線同士の成す角が小さい画素でラベリングを行い、同じラベルのみで最適化を行う方法を用いる。ラベリングは 4 近傍で行う。画素の場所の記号として図 17 に示すように、 $x$  を注目画素とした場合、上の画素を  $x_{up}$ 、左の画素を  $x_{left}$  とする。以下に、本手法で用いるラベリングのアルゴリズムを述べる。なお、ここで

は物体上の法線が推定されている画素について、図 16 に示すように左上から順に  $u$  軸方向へ走査を行うものとする。

1. (a)  $x_{up}$ ,  $x_{left}$  がどちらもラベル付けされていない場合、新たなラベルを付ける。
  - (b)  $x_{up}$  がラベル付けされており、 $x_{left}$  にラベル付けされていない場合
    - $x$  と  $x_{up}$  の法線の成す角が閾値より小さい場合、 $x_{up}$  のラベルを与える。
    - $x$  と  $x_{up}$  の法線の成す角が閾値以上の場合、新たなラベルを与える。
  - (c)  $x_{up}$  がラベル付けされてなく、 $x_{left}$  にラベル付けされている場合
    - $x$  と  $x_{left}$  の法線の成す角が閾値より小さい場合、 $x_{left}$  のラベルを与える。
    - $x$  と  $x_{left}$  の法線の成す角が閾値以上の場合、新たなラベルを与える。
  - (d)  $x_{up}$ ,  $x_{left}$  がどちらもラベル付けされている場合
    - $x$  と  $x_{up}$ ,  $x$  と  $x_{left}$  の法線の成す角が閾値以上の場合、新たなラベルを与える。
    - $x$  と  $x_{up}$  の法線の成す角が閾値より小さく、 $x$  と  $x_{left}$  の法線の成す角が閾値以上の場合、 $x_{up}$  のラベルを与える。
    - $x$  と  $x_{up}$  の法線の成す角が閾値以上で、 $x$  と  $x_{left}$  の法線の成す角が閾値より小さい場合、 $x_{left}$  のラベルを与える。
    - $x$  と  $x_{up}$ ,  $x$  と  $x_{left}$  の法線の成す角が閾値より小さい場合、 $x_{up}$ ,  $x_{left}$  のラベルのうち小さい方のラベルを与える。ルックアップテーブルの内容を調べ、その内容が大きい方のラベルに等しい場合小さい方のラベルに置き換える。
2. 以上の処理を全画素で行った後、ルックアップテーブルに従いラベルの値を更新する。

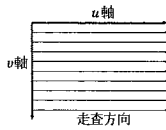


図 16: 走査方向



図 17:  $x$  を注目画素とした場合の記号

### 3.4.5 初期値の設定

最適化を行う場合、局所解を避けるために正解の近傍に初期値を設定する必要がある。そのため本手法では特徴点の 3 次元座標と法線情報を用いて初期値の設定を行う。

まず初期値として、文献 [3] の方法で得られた特徴点の奥行き  $z$  はその値を用いる。そのほかの点に関しては、隣接する点を直線に近似して  $u$  軸、 $v$  軸の隣の値を次の式で表す。

$$z_{u+1,v} = z_{u,v} - p_{u,v} * \Delta x \quad (21)$$

$$z_{u-1,v} = z_{u,v} + p_{u,v} * \Delta x \quad (22)$$

$$z_{u,v+1} = z_{u,v} - q_{u,v} * \Delta y \quad (23)$$

$$z_{u,v-1} = z_{u,v} + q_{u,v} * \Delta y \quad (24)$$

この式により、 $v$  軸優先と  $u$  軸優先の 2 つの方法で初期値を与える (図 18)。 $v$  軸優先は画像中心に近い特徴点から  $v$  軸のそれぞれの方向に式 (23), (24) で  $z$  の初期値を与える。そして求まった点から  $u$  軸のそれぞれの方向に式 (21), (22) で初期値を与える。ただし、特徴点があり 3 次元座標が分かっている場合はその値に更新する。また  $u$  軸優先では逆に  $u$  軸から推定する。

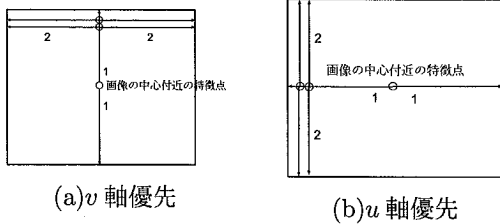


図 18: 初期値の与え方

### 3.4.6 形状復元の処理の流れ

以上をまとめると処理の流れは以下の通りである。

#### 1. 初期値の設定

特徴点の 3 次元座標と法線情報を用い、隣合う画素を直線で近似し、初期値を設定する。

#### 2. ラベリング

法線情報をもとに、法線の成す角が小さいものでラベリングを行い、面に分割する。

#### 3. 最適化

ラベリングの結果より、ラベル別に以下を行う。

- 端点周辺の法線の設定

端点の外側の 1, 2 画素の法線を、端点の法線と同じ値にする。

- 共役勾配法による奥行き  $z$  の算出

特徴点の 3 次元座標と法線情報を用いた共役勾配法による最適化により奥行き  $z$  を算出する。ただし端点の内側 1, 2 画素と外側 1 画素については別の最適化の関数を用いる。

## 4 実験

### 4.1 シミュレーションによる実験

#### 4.1.1 球のシミュレーションデータを用いた実験

図 19 のような画像について正しい法線情報、特徴点の 3 次元座標が得られたとする。対象物体は、1 画素を 1 とする  $150 \times 150$  の平面に中心が (75,75)、半径 50 の半球をおいたものである。この球に対して各画素の法線情報、ランダムな 100 点で特徴点の 3 次元座標が正しく得られたとして、データを与える。このデータに対して、3.4 節の手法で 3 次元形状復元を行った結果を図 20 に、真値を図 21 に示す。特徴点の最適化の更新については信頼できる 3 次元座標のため更新は行わず、初期値は  $v$  軸優先で与えた。

図 20 より、端点については一部真値と離れた点があるが、球の形状については復元できている。そのため、正しい法線情報と特徴点の 3 次元座標が与えられた場合、適切に形状が復元できることが確認できた。

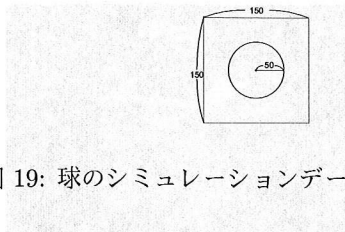


図 19: 球のシミュレーションデータに用いる画像

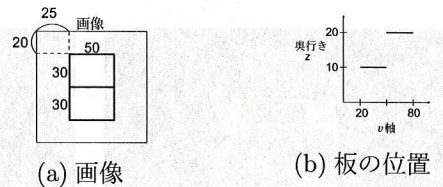


図 22: 2枚の板のシミュレーションデータに用いる画像と位置



図 20: 球のシミュレーションデータによる形状復元結果

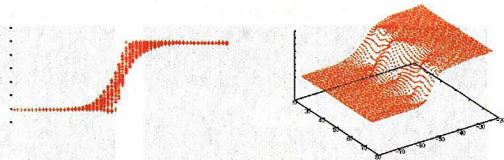


図 23: 板のシミュレーションデータによる形状復元結果

#### 4.1.2 2枚の板のシミュレーションデータを用いた実験

図 22 のように、位置が違う 2 枚の板を対象として撮影した画像について、各画素の法線情報、ランダムに与えた 100 点の特徴点の 3 次元座標の真値が得られたとしてデータを与えた。特徴点は正しい値を与えているため、最適化の更新は行わないものとする。また初期値は  $\psi$  軸優先で与えた。3.4 節の手法による形状復元結果を図 23 に示す。

図 23 より、板の境目では適切な推定になっていないが、2 枚の板の形状が復元できている。このシミュレーションデータでは法線方向はすべて同じであるので、ラベリングによる面の分割手法では 2 つの板に分割することができない。それに対して特徴点の 3 次元座標を与えることで法線情報には現れない不連続な面の形状においても復元できている。

#### 4.1.3 2個の立方体のシミュレーション画像を用いた実験

まず 3.2 節の手法で特徴点の 3 次元座標を得るためのシミュレーション画像を作成するため、図 24 に示すように物体とカメラを配置した。



図 21: 球の真値

対象物体は  $20 \times 20 \times 20$  の立方体の上に  $10 \times 10 \times 10$  の立方体をおいたものである。物体のテクスチャとしては下の立方体には石垣のような画像を、上の立方体には赤いレンガのような画像を張っている。物体の中心の 10 度上方からカメラを物体を中心として円軌道上、反時計回りに 10 度ずつ移動し撮影した全周囲画像 36 枚を作成した。得られたシミュレーション画像の一部を図 25 に示す。3.4 節の手法により得られた特徴点の 3 次元座標の算出結果を図 26 に示す。

図 26 を見ると、物体上面に飛びだしている点はあるが、物体の形状を適切に復元できていることが分かる。

次に法線情報を得るため、第 1 画像と同じ撮影状況において光源位置を変えて撮影した 4 枚の画像を作成した。作成した画像を図 27 に示す。この 4 枚の画像から 3.2 節の手法で推定した法線を 20 画素おきに表示したものを図 28 に示す。図 28 を見ると、面毎に適切な法線方向が得られていることが分かる。

以上の処理より得られた特徴点の 3 次元座標、法

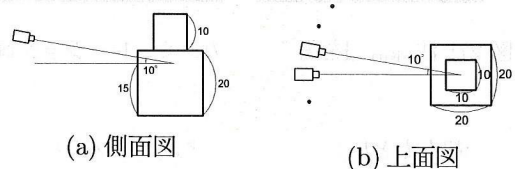


図 24: 物体形状とカメラの位置関係



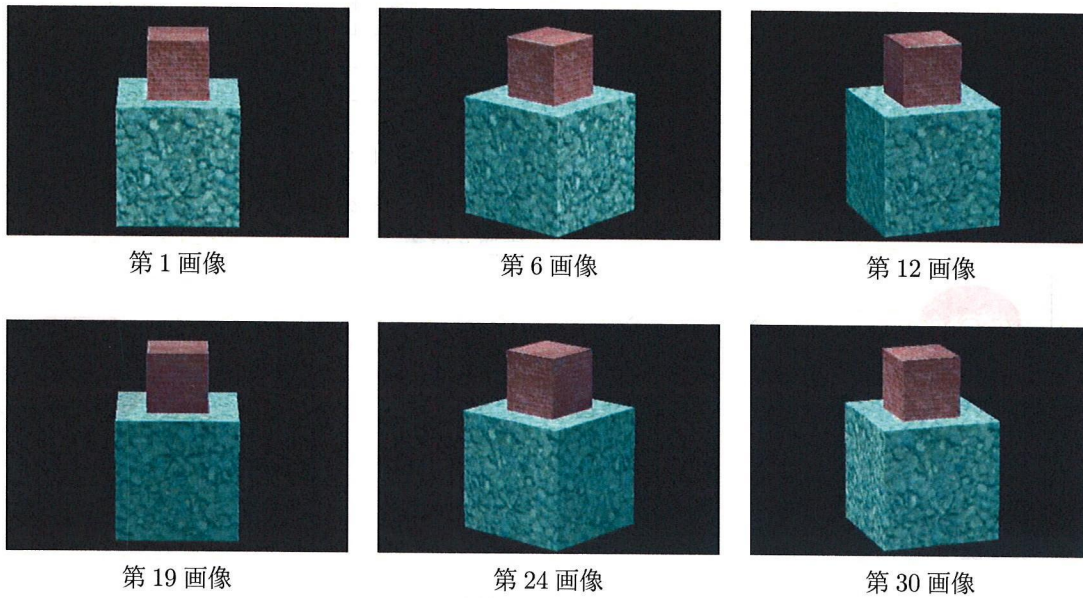


図 25: 特徴点对応による復元に用いるシミュレーション画像列の一部

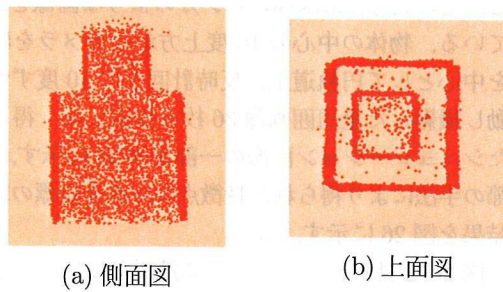


図 26: 特徴点对応による 3 次元座標の算出結果

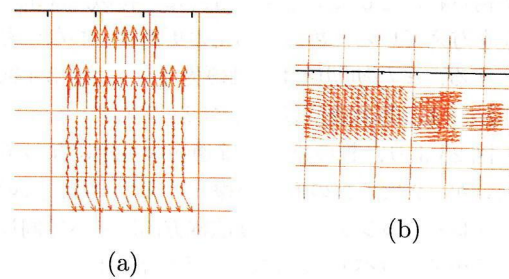


図 28: 法線の推定結果

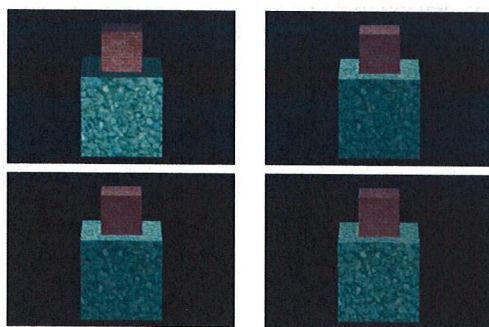


図 27: 法線情報を得るためのシミュレーション画像

いては、3次元座標として信頼できない点のため、一定回数特徴点の更新を行わず最適化した後、特徴点も含めた最適化を行った。また初期値は物体形状から、 $v$  軸優先で与えた。

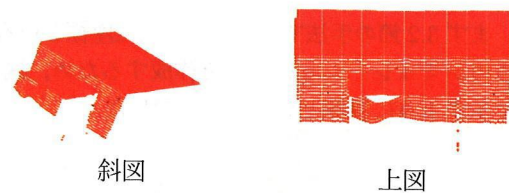


図 29: 本手法による形状復元結果

線情報をもとに 3.4 節の統合アルゴリズムを適用した結果を図 29 に示す。特徴点の最適化の更新につ

図 29 より、一部の点と上の立方体の上面では歪みが見られる。これは特徴点の 3 次元座標の中で適切な位置にない点が影響している。しかし文献 [3]

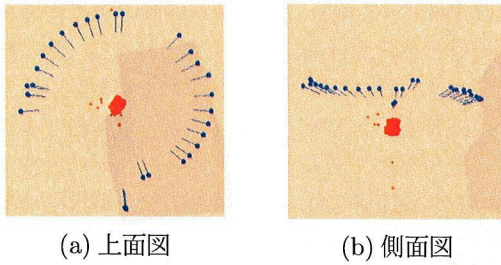


図 31: 特徴点とカメラの 3 次元座標算出結果

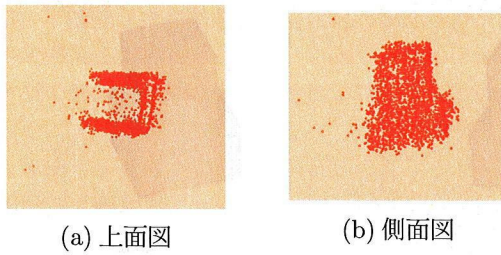


図 32: 特徴点の 3 次元座標

に比べて細かな形状が復元できており、また立方体の形状についても復元できていることが確認できた。また立方体の前面と上面は 90 度ではない。これは  $u$  軸、 $v$  軸と  $z$  軸のスケールがっていないためである。そのため今後スケールを合わせ、形状復元結果を正しく見えるようにする必要がある。

## 4.2 実画像を用いた実験

次に実画像による実験を行った。対象物体はコルクの直方体の上に一回り小さいコルクの立方体をのせたものである。対象物体の全周囲 30 枚と第 1 画像のカメラ位置・姿勢において異なる光源状況下の 4 枚の画像をデジタルカメラ (Canon EOS 30D) により撮影した。このとき焦点距離は一定である。また光源方向は既知である。得られた画像の一部を図 30 に示す。また画像は背景は全て白としている。3.3 節の手法により得られた特徴点とカメラの 3 次元座標の算出結果を図 31 に、特徴点の 3 次元座標を拡大したものを図 32 に、3.2 節により得られた法線の推定結果を 5 画素おきに表示したものを図 33 に示す。ここで周囲の円弧上の線がついた点はカメラ位置を表し、線はカメラの光軸を表している。また点のみのものは特徴点の 3 次元座標を表している。

カメラ位置はほぼ同じ間隔で撮影したが図 31 の

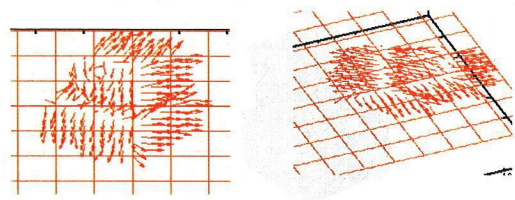


図 33: 法線の推定結果



図 34: 実画像による形状復元結果 1

推定したカメラパラメータでは一部大きく離れている所があり、適切に推定できていない。しかし図 32 より、数点飛びだしている点はあるが、おおまかに形状復元はできている。また図 33 より表面法線は適切に推定できている。この 2 つの推定結果をもとに 3.4 節を用いて統合した結果を述べる。まず本手法をそのまま用いた場合、ラベリング結果の多くは少数の点からなるラベルに分割された。少数の点では非線形最適化がほとんど行われず、ほぼ初期値と変わらない結果となった。そのため、ラベリングを行わず、全体で最適化を行った結果で、5 画素おきに表示したものを図 34 に示す。特徴点の最適化の更新では 3 次元座標は信頼できないため、一定回数更新を行わず最適化を行った後、特徴点も含めた最適化を行った。また初期値は物体形状から  $v$  軸優先で与えた。

図 34 より、大まかな形状は復元できているが、上の立方体の上面では面が広がり、適切に形状が復元できていない。これは、3.2 節の手法で得た特徴点の 3 次元座標において推定に間違いがあり、本来の 3 次元座標から離れた位置にあるためその点が影響し適切に形状が推定できていないと考えられる。この問題を解決するため最初から特徴点で得られた 3 次元座標も更新した。その結果で、5 画素おきに表示したものを図 35 に示す。また法線情報のみで復元した結果と比較するため、1 点の特徴点のみ 3 次元座標を与えて復元した結果で、5 画素おきに表示したものを図 36 に示す。





図 30: 実画像による実験に用いる画像列

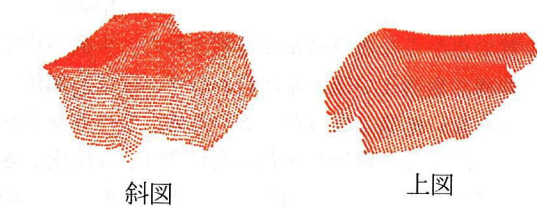


図 35: 実画像による形状復元結果 2(特徴点を初期値とする)



図 36: 実画像による形状復元結果 3(法線情報のみ)

図 35 より特徴点の 3 次元座標を初期値として与え、更新を行うことで適切に形状が推定できていることが確認できた。法線情報のみを用いた図 36 では上の立方体の上面では正方形にはなっていないが、図 35 では正方形に近い形状になっており、より適切に形状を復元できている。そのため特徴点を用いた初期値の設定は有効であるといえる。

## 5 まとめ

本論文では陰影情報により得られた法線情報と特徴点对応により得られる 3 次元座標を統合することで高精細な 3 次元形状を復元する手法を示した。本手法では、まず画像の陰影情報をもとにして得た法線情報と特徴点对応による特徴点の 3 次元座標を得る。次に法線情報をもとにラベリングを行い、面毎に分割する。分割した面毎に法線情報と特徴点の 3 次元座標を用いた最適化により奥行きを算出し 3 次元形状を復元した。

シミュレーションデータを用いた実験により、法線情報と特徴点の 3 次元座標の正しい値が得られた場合、連続な面による形状や不連続な面を含む形状に対して、適切な形状復元が行えることを確認した。またシミュレーション画像による実験により不

連続な面を含む形状に対して、文献 [3] の手法より点数の多い、高精細な形状復元が行えることが確認できた。実画像を用いた実験では、法線方向が同じ面でも一様とならず、ラベリングによる面分割が適用できなかった。また特徴点の 3 次元座標の推定の誤差が形状復元に影響した。しかし特徴点の 3 次元座標を初期値として与え最初から最適化を行うことで適切な形状を復元できた。この手法により適切な形状を復元できることも確認した。しかしこの手法では、不連続な面の場合は適切な形状復元が得られないと考えられる。

今後の課題として、実画像などに対しても不連続な面や、法線情報や特徴点の 3 次元座標に誤差が含まれる場合も適切な形状復元が行えるようにすることが挙げられる。また  $u$  軸、 $v$  軸、 $z$  軸のスケールを合わせ、形状復元結果が正しく見えるようにする必要はある。今後は、全周囲に対する復元を行い完全な 3 次元形状復元を行うことや法線情報を用いて表面モデルを推定することが考えられる。

## 参考文献

- [1] 田村秀行: “コンピュータ画像処理”, 株式会社オーム社, (2002).
- [2] 沈 黎, 町田 貴史, 竹村 治雄: ” 効率的なフォトメトリックステレオ法による 3 次元形状と表面反射特性の推定”, 画像の認識・理解シンポジウム (MIRU2005), pp.1305-1312.
- [3] 保田 和隆, 椋木 雅之, 浅田 尚紀:” 大局的幾何学制約を用いた特徴点追跡に基づく画像列からの 3 次元モデル生成 “, Vol.47 No.SIG 10(CVIM 15) 情報処理学会論文誌: コンピュータビジョンとイメージメディア July 2006 pp.96-106.
- [4] 佐藤 智和, 神原 誠之, 横矢 直和, 竹村 治雄: ”マルチベースラインステレオ法を利用した動画像からの屋外環境の三次元モデル化” , バーチャルリアリティ学会論文誌, No.7, Vol.2, 2002.
- [5] 杜菲, 岡部孝弘, 佐藤洋一, 杉本晃宏, ”複雑照明下における運動物体の反射特性の推定”, 電子情報通信学会論文誌, Vol.J88-D-II, No.8, pp.1460-1468, August 2005.
- [6] C. Harris and M. Stephens: “A Combined Corner and Edge Detector,” *Proc. Alvey Vision Conf.*, pp.147-151, 1988.
- [7] R.T.Frankot and R.Chellappa, “A method for enforcing intergrability in shape from shading algorithms”, *IEEE Transactions on pattern Analysis and Machine Interlligence*,10(1988)439-451.
- [8] A. Amano, T. Migita and N. Asada: “Stable Recovery of Shape and Motion from Partially Tracked Feature Points with Fast Non-linear Optimization,” *Proc on Vision Interface*, pp.244-251, 2002.
- [9] J.D.Foley, A.Dam, S.K.Feiner, J.F.Hughes, 佐藤 義雄 訳 “コンピュータグラフィクス理論と実践”, オーム社, ISBN4-274-06405-0, 2001.
- [10] B.T.Phong, “Illumination for computer generated pictures”, In *Communications of the ACM*, Vol.18, No.6, pp.311-317, 1975.
- [11] 岡本 崇弘: ” バンプモデルを用いた実物体の反射特性のモデル化とパラメータ推定”, 広島市立大学大学院情報科学部 知能情報システム工学科 修士論文, 2004.
- [12] Yvan G.Leclerc and Aaron F.Bobick “The Direct Computation of Height from Shading” *IEEE Conf. Comput. Vision Patt. Recogn. (Hawaii)*, May 1991, pp. 552-558.
- [13] 右田 剛史, 天野 晃, 浅田 尚紀 “3 次元形状・運動復元のための高速非線形最適化計算法” *情報処理学会論文誌*, Vol.44, No.11, pp.2864-2872(2003).





## 第3章

### 画像特徴の高精度追跡

- 大局的幾何学制約を用いた特徴点追跡に基づく画像列からの3次元モデル生成
- 3D Model Generation from Image Sequences using Global Geometric Constraint
- 奥行き制約による対応点探索を用いた画像列からの3次元モデル生成
- 線分要素の自動対応付けに基づいた画像列からの疎な3次元形状復元

# 大局的幾何学制約を用いた特徴点追跡に基づく 画像列からの3次元モデル生成

## 3D Model Generation from Image Sequences using Global Geometric Constraint

### あらまし

本論文では、位置・姿勢が未知のカメラを用いて物体の全周囲を撮影した画像列から、3次元モデルを生成する一連の手順を提案する。3次元モデルの生成には、画像間に対応する特徴点を利用する。この場合、相関法のような局所情報のみによる特徴点追跡では、対応誤りが発生する。本手法ではまず、アフィン変換を用いて対応付けた画像全体の位置関係という大局的幾何制約（アフィン制約）を導入し、対応誤りが少ない特徴点追跡を行う。次に、再投影誤差最小化手法を直接適用することにより、対応点の誤差にロバストに形状復元を行う。さらに、復元物体の全体構造に基づく大局的幾何制約（奥行き制約）を用いた特徴点追跡と形状復元を繰り返す。これにより、特徴点の3次元位置を安定して復元でき、テクスチャを付加した高精細な3次元モデルが生成できることを示す。

### Abstract

This paper describes a method for generating a three-dimensional model from an uncalibrated image sequence taken around an object. Our method is based on feature tracking and minimization of re-projection errors. To cope with mis-matches in the result of feature tracking, we introduce two types of global geometric constraints. The one is “affine constraint” which imposes the positional relationship between pixels on the images. The other is “depth constraint” which imposes the three-dimensional structure of the object. First, we use the affine constraint to reconstruct the object roughly and then we refine the feature tracking and shape reconstruction using the depth constraint. Experimental results show that our method can automatically generate accurate three-dimensional models from real image sequences.

## 1 はじめに

実物体の3次元モデル化は、仮想空間におけるリアルな物体表現として利用されるだけでなく、文化財のデジタルアーカイブなど多様な目的に用いられる重要な技術である [1]。最近では、レーザレンジセンサを用いて高精細な3次元モデルを生成することが行われているが、専用の計測装置が必要であり、大きさや表面特性などいくつかの条件を満足する対象に限られるという性質がある。このため従来から、迅速かつ手軽に形状を生成できるシステムが渴望されており、コンピュータビジョン技術の応用が試みられている。

その一つとして、多視点画像からカメラと対象物の相対運動及び対象物の3次元形状を復元する画像列からの3次元形状復元（Structure from Motion）を用いた3次元モデル生成が研究されている [2, 3, 4, 5, 6, 7, 8]。

画像列からの形状復元を用いた3次元モデル生成では、まず、視点の異なる画像間に対応する点を追跡し、カメラの位置・姿勢と3次元形状の復元を行った後、最終的な3次元モデル生成が行われる。

特徴点追跡においては、Tomasiら [9] がビデオカメラで撮影した視差の微小な連続画像間に対し、輝度勾配を用いた特徴点追跡手法を提案している。

また, Zhangら [10] や金澤・金谷 [11] は, 視差の大きな画像間に対し, 画像情報のみからエピポラ方程式を求め, 特徴点のランダム選択と局所対応の投票などにより, 特徴点の対応付けを行う手法を提案している. しかし, これらの手法では, 物体の見え方の変化や隠蔽, フレームアウトなどにより, 大きな特徴点の対応誤りを生じたり, 特徴点の追跡結果が得られない場合がある.

最終的な3次元モデルの生成には, マルチベースラインステレオ法 [12] や距離推定 [13], 多視点解析 [14] などの手法が用いられる. これらの処理は, カメラ位置・姿勢が正確に求められていることが処理の前提となっているが, 一般に, 画像列からの形状復元により求められるカメラ位置・姿勢には誤差が含まれている. 特に, 特徴点の対応誤りに起因する誤差の影響は大きく, 3次元モデルを安定して生成することは困難であった.

このように従来の研究では, 個々の技術については多くの研究がなされているが, 大きな視差を含む数十枚の全周画像列から3次元モデルを生成する一連の処理を構築するという観点での議論は十分なされていない. この一連の処理では, 各段階に従来手法を適用することが考えられるが, 前提としている条件が異なるため, それを組み合わせただけでは適切に3次元形状が復元できない. そのため, 途中段階の処理に含まれる誤差を許容しつつ, 新しい制約を追加していくことで段階的にモデル形状を復元する方法を考案する必要がある.

本研究は, 物体の全周囲を撮影した画像列から, 大局的幾何制約により特徴点追跡を行い, 再投影誤差最小化原理に基づき安定に3次元モデルを生成する一連の手法を提案する. 本手法では, まず, 2画像間の変形をアフィン行列で表せると仮定し, 変形した画像における画素の近傍領域に特徴点の対応付けを限定するという大局的幾何制約(アフィン制約)により, 初期的な特徴点追跡を行う. この追跡結果をもとに, 非線形最適化計算により, 特徴点位置とカメラ位置・姿勢を推定する. 得られた特徴点位置とカメラ位置・姿勢を基に, 特徴点の存在する奥行き範囲を限定するとい

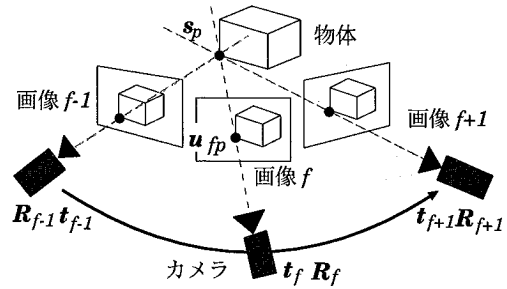


図 1: 形状復元とカメラ位置・姿勢の推定

う大局的幾何制約(奥行き制約)を導入し, 限定した範囲内を画像上で探索することにより, より信頼性の高い特徴点追跡を行う. 以上の処理を繰り返すことにより, 明らかな誤対応を確実に排除し, 精度を向上させ, 幾何学的に正しい3次元形状復元を行う. 最終的に, 信頼できる特徴点のみから構成されたテクスチャマッピングを行った3次元モデルを生成する.

## 2 再投影誤差最小化による形状復元

形状復元においては, 物体は  $P$  個の特徴点で表現される (1). 各特徴点は, 物体とカメラを含む共通の座標系(世界座標系)上にあり, 固有の3次元座標  $s_p$  ( $p = 1 \dots P$ ) を持つ. これらの点は最大  $F$  枚の画像上に投影され観測される. 第  $f$  画像 ( $f = 1 \dots F$ ) 上で観測された, 第  $p$  特徴点の座標を  $u_{fp}$  とする. 特徴点  $p$  が画像  $f$  で観測されていない場合  $u_{fp}$  は定義されない.

各画像を撮影したカメラの位置・姿勢は, 世界座標系における3次元の回転と並進  $(R_f, t_f)$  で表される. 投影モデルは透視投影とし, 焦点距離は全画像を通じて共通の既知の値  $l$  とする.

これらのパラメータを用いて, 形状復元問題は

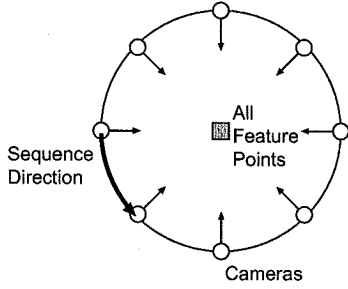


図 2: 注視回転による初期値

次のように定式化される [15, 16].

$$\begin{aligned} \arg \min_{\mathbf{x}} E(\mathbf{x}) & \quad (1) \\ E(\mathbf{x}) & := \sum_{(f,p)} |\mathcal{P}[R_f \mathbf{s}_p + \mathbf{t}_f] - \mathbf{u}_{fp}|^2 \\ \mathbf{x} & := (\mathbf{s}_p, \mathbf{q}_f, \mathbf{t}_f \text{ を並べたベクトル}) \\ \text{ただし, } \mathcal{P}[x, y, z]^T & := \frac{1}{z}[x, y]^T \end{aligned}$$

ここで  $R_f$  は自由度 3 の回転行列であり,  $\mathbf{q}_f$  は  $R_f$  のパラメータを並べたベクトルである. また, 特徴点の画像座標  $\mathbf{u}_{fp}$  とその画像間での対応は予め与えられているものとする.

この式は, 推定した形状  $\mathbf{s}_p$  を推定したカメラパラメータ  $R_f, \mathbf{t}_f$  で投影した場合の投影座標  $\mathcal{P}[\cdot]$  と, 観測値  $\mathbf{u}_{fp}$  との誤差 (これを再投影誤差と呼ぶ) の 2 次ノルムの総和が最小となることを要求するものであり, 観測座標に等方性正規分布の誤差を仮定した場合の最尤推定を与える式である. 即ち, 正規分布とみなせるような微小な誤差に対しては, ロバストに形状復元が行えることになる.

式 (1) は, 非線形最適化問題となる. 一般に, 非線形最適化計算では, 繰り返し処理により解を逐次更新する反復法が用いられる. 計算法としては, Levenberg-Marquardt 法などの任意の汎用非線形最適化手法が利用できるが, 問題の規模が大きい場合, より計算量の少ない, 前処理付き共役勾配法 [17] が適している.

非線形最適化計算においては, 初期値の設定が重要となる. 本研究では, 物体の周囲を物体とほぼ同じ水平面内で一方向に移動しながら撮影した画像列を用いることを想定する. この撮影方法を,

物体を中心にカメラが等速円運動をしながら物体を観測する注視回転によりモデル化する (2). この場合, 初期値を数式で表すと次式のようなになる.

$$\begin{cases} R_f = R_{\omega_f}^y, \mathbf{t}_f = (0, 0, r)^T & \text{for all } f \\ \mathbf{s}_p = (0, 0, 0) & \text{for all } p \end{cases} \quad (2)$$

ここで,  $R_{\omega_f}^y$  は,  $y$  軸 (2 の紙面鉛直方向) 回りの角度  $\omega_f$  の回転を表す行列とする. また,  $r$  はカメラの円軌道の半径であるが, 計算空間の 3 次元座標系には物理単位が与えられていないため,  $r = 1$  として良い. 対応点追跡が正しく行われており, 物体のほぼ全体が画像内に撮影されている場合, この初期値からほとんどの場合で最適な形状復元が行える [18].

### 3 大局的幾何制約の導入

再投影誤差最小化による形状復元は, 正規分布とみなせるような微小な誤差に対してはロバストであるが, 統計的に外れ値とみなされる大きな特徴点の誤対応には対処できない.

一般に誤対応の除去には, Fischler ら [19] が提案した RANSAC や Rousseeuw [20] が提案した最小メジアン法 (LMedS 推定) が用いられる. これらの手法では, 選択する対応候補の中に十分な数の正しい対応が存在しなければならない. しかし, 視差の大きな画像間や, 類似したテクスチャの多い画像が対象の場合は, 特徴点の対応付け結果の正解率が下がり, これらの手法で推定可能な精度を満たさない.

一方, 画像間で対応するエピポーラ線を求め, 特徴点の探索範囲をエピポーラ線上に限定することで, 誤対応を軽減する手法も用いられている. しかし, 画像列からの形状復元により推定されたカメラ位置・姿勢を利用する場合, 推定誤差が含まれるため, 必ずしも対応点がエピポーラ線上に存在しているとは限らない. これに対処するには, エピポーラ線の近傍に対しても対応点探索の範囲を拡大する必要がある [22, 21] が, 探索範囲を広げすぎると誤対応が生じる可能性が高まる.

本研究では、数十枚の画像列を処理するため、明らかな誤対応を確実に排除する必要がある。そのために、2種類の大局的幾何制約を導入する。

特徴点追跡の最初の段階では、利用できるのは画像情報のみであるため、まず画像そのものを直接対応させることで大局的制約を与える。即ち、2画像間の見え方の変化を幾何学的変換により近似し、画像全体が最もよく対応づけられる変換を求める。画像内の各画素は、この幾何学的変換により変換された位置の近傍に対応点が存在すると仮定し、特徴点の対応づけをその範囲に限定する。画像対応のための幾何学的変換は、視差による見え方の変化をある程度表現できる自由度が必要であるが、過度の自由度は計算が不安定となる欠点がある。そこで、本研究では自由度6のアフィン変換を用いる。アフィン変換を用いた画像間の対応付けに基づく幾何制約をアフィン制約とよぶ。

アフィン制約により、特徴点对応における大きな外れ値は除外できるが、アフィン変換は画像間の変形を完全には表現できないため、検出できる特徴点数が少ない上、多少の誤差も存在する。より精度良く3次元形状を復元するために、特徴点追跡結果に基づき、一旦形状復元を行う。得られた3次元形状とカメラ位置・姿勢を用いると、画像内の各点における物体までの奥行きを求めることができる。この結果は誤差を含むが、一定の範囲を考慮することにより、誤差を許容しつつ特徴点の対応範囲を限定することができる。この制約を奥行き制約とよぶ。

RANSACやLMedS推定を用いた方法やエピポーラ線上の特徴点探索は、数点の特徴点の局所的な対応付け結果に基づいている。これに対して、本手法では画像全体あるいは物体の3次元構造全体をさらに利用することにより、適切に特徴点の対応範囲を限定することができ、より安定して形状復元を行うことが可能である。金澤・金谷[23]は、検出した特徴点に対して、局所相関、対応点のフロー、射影変換の一致度を確信度として逐次的に追加しながら、大域的な整合がとれるよう対応づけを更新し、最終的にエピポーラ拘束により対応付けを行う手法を提案している。一方、本手

法は、アフィン制約を用いた初期復元の結果を利用し、形状復元と対応点探索を交互に行いながら全体を最適化することにより、モデル形状に含まれる誤りを除去する方法となっている。

## 4 3次元モデル生成

### 4.1 処理の流れ

3次元モデル生成の処理の流れを、3に示す。物体の全周囲を20~40枚程度で撮影した画像列が与えられる。撮影したカメラの焦点距離は既知とする。まず、アフィン制約により、特徴点の追跡を行う。次に、追跡結果から再投影誤差最小化により形状とカメラ位置・姿勢を復元する。この際、復元精度を向上するために、誤対応とみなせる点の除去を行う。得られた復元結果を利用し、奥行き制約に基づき特徴点追跡を行う。この際、画像の各点の奥行きを求めるために、点群で表現された物体形状に、ドロネー法により3角面を生成する。奥行き制約を更新しながら、形状復元と特徴点追跡を繰り返すことにより、段階的に3次元形状の精度を向上させていく。特徴点追跡において、追跡される特徴点数がほぼ一定に収束した段階で、得られた3次元形状に対してテクスチャ画像を選択、付加することにより3次元モデルを生成する。

### 4.2 アフィン制約を用いた特徴点の対応付け

アフィン変換行列

$$A = \begin{bmatrix} a & b & t_u \\ c & d & t_v \end{bmatrix}$$

を用いると、画像 $I$ 上の座標 $(u, v)$ と画像 $I'$ 上の座標 $(u', v')$ の対応は、次式により表される。

$$[u' \ v']^T = A[u \ v \ 1]^T \quad (3)$$

2画像間の最適アフィン変換行列 $A$ は、対応する画素値の差の2乗和が最小となるように次式に

よって求める.

$$\arg \min_A \sum_{(u,v)} |I(u,v) - I'(u',v')|^2 \quad (4)$$

ここで,  $I(u,v), I'(u',v')$  はそれぞれ, 画像  $I, I'$  の座標  $(u,v), (u',v')$  での画素値とする. 上記の計算は, まず並進成分  $[t_x, t_y]^T$  の最適値を 1 画素毎の全探索により求め, 次にその値を初期値として, 6 個すべてのパラメータを勾配法により求める 2 段階の探索により行った. なお, 局所解を避けるため画像を平滑化し低周波成分によるマッチングを行う.

実際の手順は, 下記の通りである (4).

1. 画像  $I, I'$  の両方において, Harris の手法 [24]<sup>1</sup> を用いて特徴点を検出する.
2. アフィン変換前の画像において, 画像  $I$  の 1 つの特徴点  $P$  に対して, 画像  $I'$  の特徴点の中で, 周囲画素同士の画素値の残差 2 乗和 (SSD) を評価基準として, 最も類似しているものから順に  $m$  個の点  $Q_1 \sim Q_m$  を選択しておく.
3. 2 画像間で求めたアフィン変換行列を用いて, 画像  $I$  の特徴点  $P$  を画像  $I'$  上に変換する. 変換した点を  $P'$  とする.
4.  $Q_1 \sim Q_m$  のうち,  $P'$  の近傍領域に含まれるものがあれば, それを対応点とする. 複数あれば, その中で SSD の評価基準により最も類似している点と対応づける. 近傍領域に含まれる点が無ければ, 点  $P$  の対応点はないものとし, 特徴点から除外する.

以上の (1)~(4) の処理を, 画像  $I$  の全ての特徴点に対して順に行う. この際, 画像  $I$  の特徴点  $P$  は, 対応する特徴点の第 1 候補  $Q_1$  の SSD による評価値が最もよい (類似している) ものから順に選択する. また, 特徴点  $P$  と対応づけられた特徴点  $Q_{i^*} (i^* \in 1 \dots m)$  は, 以降の対応付けからは除外する. この手法は, 従来から行われてい

<sup>1</sup><http://www.img.tutkie.tut.ac.jp/programs/index.html> のプログラムを利用した.

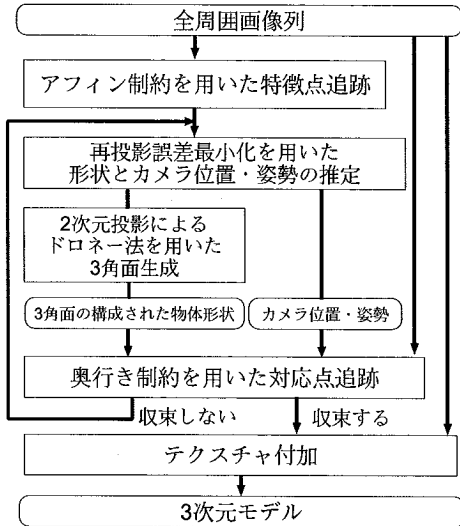


図 3: 提案手法の処理の流れ

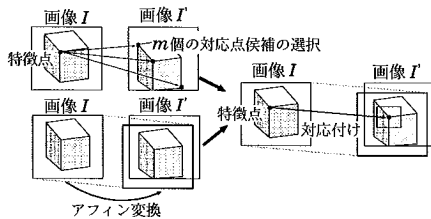


図 4: アフィン制約を用いた特徴点の対応付け

る貪欲法に、アフィン制約を追加したものとなっている。これにより、大局的な幾何制約を考慮した特徴点追跡を行う。

#### 4.3 再投影誤差最小化による形状復元

前節の処理によって得られた画像列全体の特徴点对応を用いて、物体の形状復元と各画像を撮影したカメラの位置・姿勢パラメータ推定を行う。これは「運動からの形状復元」問題として多くの研究がなされているが、本研究では文献 [17] の再投影誤差最小化原理に基づく方法を用いて物体形状とカメラ位置・姿勢の同時推定を行う。

ただし、画像情報のみを利用して対応付けた特徴点は、一般に追跡誤りを含んでいるため、誤った特徴点に対処する必要がある。そのために、ロバスト推定の一環である LMedS 推定や M 推定 [25] が用いられることがあるが、非線形最適化では収束性が悪くなる問題や計算の負荷が大きくなる問題が伴う。このため、まず誤差を含む特徴点の追跡結果に対して非線形最適化を利用して、特徴点の 3 次元座標とカメラ位置・姿勢を推定する。次に、特徴点の追跡誤差によるカメラ位置・姿勢の復元誤りを補正するため、復元後に 3 次元上の特徴点を画像上へ再投影し、2 乗誤差を計算する。この評価尺度において誤差が大きな特徴点を除去する。残りの特徴点に対して再度式 (1) を適用して形状復元を行うことにより、特徴点の数は減るが、より信頼性の高い復元結果を得る。

#### 4.4 奥行き制約による形状復元の改善

前章で得られたカメラ位置・姿勢と物体構造を利用して、奥行き制約を用いてより信頼性の高い特徴点追跡を行う。そのためにまず、点群として得られた物体形状に 3 角面を生成する。

3 次元上の特徴点データを円筒面に投影し、投影した平面上で 2 次元ドロネー法を用いて 3 角面を構成する。なお、円筒面に投影するため、復元物体の重心を世界座標の原点に、上下方向を  $z$  軸として定めておく。また、円筒面座標  $(R, \theta)$  上に

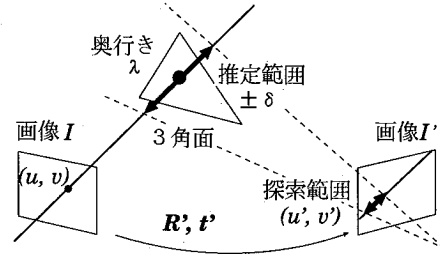


図 5: 奥行き制約を用いた特徴点の対応点探索

投影した際、適切な 3 角面を構成するために、各軸は正規化されているとする。

まず、3 次元空間上の物体を形成する  $i$  番目の特徴点  $(x_i, y_i, z_i)$  を、円筒面座標  $(R_i, \theta_i)$  で表す。

$$\theta_i = \tan^{-1} \frac{y_i}{x_i} \quad R_i = z_i \quad (5)$$

ここで、 $\theta_i$  は  $[-\pi, \pi]$  の値をとるが、3 次元空間では  $\theta_i = -\pi$  と  $\theta_i = \pi$  は同じ位置を表しているため、 $\theta = -\pi$  付近に投影された特徴点を  $\theta = \pi$  付近にも投影する。次にこの平面においてドロネー法により 3 角面を作成する。この際、 $\theta_i = -\pi, \pi$  付近の重複した 3 角面は削除する。以上の処理により構成された特徴点の接続関係を、もとの 3 次元空間での点群に当てはめることにより 3 角面の構成を行う。

3 角面が生成されると、各画像上の画素の奥行きは、5 に示すようにカメラ位置とその画素を通る直線が 3 角面と交差する奥行き  $\lambda$  の周辺に位置すると考えられる。このとき、画像  $I$  の座標  $(u, v)$  に対応する画像  $I'$  の探索位置  $(u', v')$  は、次式の範囲に制限できる。

$$\begin{bmatrix} u' \\ v' \end{bmatrix} = \mathcal{P} \left[ R' \hat{\lambda} \begin{pmatrix} u/l \\ v/l \\ 1 \end{pmatrix} + t' \right] + \begin{bmatrix} 0 \\ \alpha \end{bmatrix}$$

$$\text{ここで } \hat{\lambda} \in [\lambda - \delta, \lambda + \delta] \quad (6)$$

ここで、 $l$  は焦点距離、 $R', t'$  は画像  $I$  に対する第  $I'$  画像のカメラ位置・姿勢を表す。また、 $\delta$  は  $\lambda$  を中心とする探索する奥行き範囲、 $\alpha$  は誤差補



正のための垂直軸方向への探索範囲である<sup>2</sup>。 $\mathcal{P}[\cdot]$ は透視投影による投影座標を表す。

画像  $I$  上で特徴点検出を行い、式 (6) で求められる  $5$  に示す画像  $I'$  上の探索範囲を SSD により対応点探索する。これを全画像間で行い、得られた特徴点の追跡結果に対して、4.3 節の処理により形状復元する。以上の処理は、追跡される特徴点数が収束するまで繰り返す。

なお、本手法では、対象物は、特徴点を円筒面に投影したときに投影方向での奥行きを重ねることがないように凸形状をしていると仮定している。また、円筒面に投影しているため、物体の上面、下面では 3 角面が大きく歪んで投影され、ドローン法で適切な面が生成できない場合がある。本研究では、物体とほぼ同じ水平面内から撮影を行っているため、物体の上面、下面は、もともとほとんど観測されていないため、この点については、特に対処していない。

#### 4.5 テクスチャの付加

特徴点数が収束した段階で、最終的に得られた形状復元結果に対しテクスチャを付加する。まず、構成された各 3 角面を画像列中の各画像上に投影し、可視判定を行う。次に可視である画像の中で投影結果が最大面積となる画像を、3 角面の最適なテクスチャ画像として選択する。この際、同一平面上にあると判定された点群からなる 3 角面には、同一の画像をテクスチャ画像として用いることにより、違和感のないテクスチャマッピングを行う。

## 5 実験と評価

### 5.1 シミュレーション実験

仮想的な空間に、一辺 10unit と 5unit のテクスチャのある立方体を置き、6 に示す環境を設定した。ここで、1unit は、仮想的な空間における長

<sup>2</sup> $\alpha$  は、本来エビ極面に垂直な方向の対応誤差を表すパラメータであるが、本研究ではカメラが水平移動し、光軸回りの回転がないため、式 (6) を用いた。

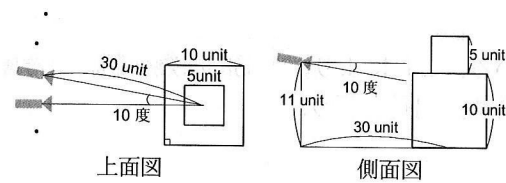


図 6: 物体形状とカメラの位置関係

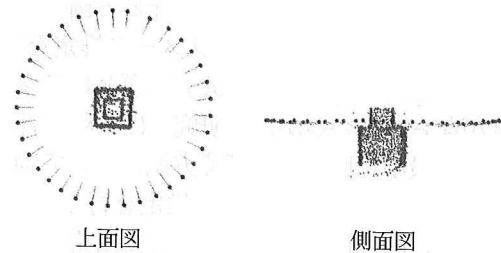


図 7: アフィン制約を用いた初期復元結果

さの基準単位とする。全周囲を 10 度刻みに撮影し、36 枚のシミュレーション画像 ( $720 \times 480$  画素) を得て実験を行った。

アフィン制約を用いて特徴点追跡を行い、形状復元した結果を 7 に示す。この実験では、各画像から特徴点を 300 点検出し、アフィン変換で対応を求めた画素から 15 画素以内の範囲で特徴点対応付けを行った。中央の密集した点が 3 次元空間中の特徴点を表し、周囲の円弧上の点がカメラの位置、カメラ位置から中央の特徴点群に向かう線がカメラ光軸を表している。

7 では、特徴点位置は一部不正確であるが、全体の大まかな形状は、復元されている。また、カメラ位置は円弧状になっており、安定に復元できている。

次に、復元された物体形状とカメラの位置・姿勢を用いて、奥行き制約により特徴点追跡を行う。奥行き制約を用いない一意に指定した奥行きの移動範囲と、奥行き制約を用いた場合の奥行きの移動範囲を画像上に投影した領域を 8 を示す。奥行き制約を用いない手法では、物体全てを含むように、奥行き  $\lambda'$  を 26.5~38.4 unit とした。画像上では 101 画素程度の探索範囲となる。また、本手法では各点毎の推定された奥行き  $\lambda$  に対し、経験



図 8: 探索範囲の比較

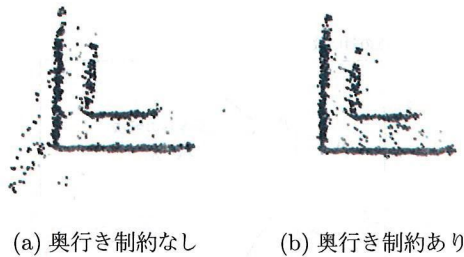


図 9: ステレオ法により求めた特徴点の 3 次元座標

的に  $\delta = \pm 1.52$  unit の範囲（画像上では 14 画素程度）とした。なお、カメラ位置・姿勢の誤差を軽減するため、式 (6) 中の  $\alpha$  を  $\pm 5$  画素として、画像の垂直軸方向へも探索を行っている。

以上の設定の下で、両手法で特徴点の対応点探索を行う。これより得られた特徴点の 3 次元座標を、9 に示す。

9(a) が奥行き制約を用いない場合の結果であり、9(b) が奥行き制約を用いた場合の結果である。なお、特徴点は一方の画像のみから 3,000 点検出し、もう一方の画像では各特徴点の探索範囲内で対応点探索を行う。奥行き制約を用いない 9(a) は、物体の前に点群が存在している。これは誤った特徴点に対応づけられたためである。本手法では探索範囲が各特徴点ごとに制限されているため、誤対応が軽減され、このような誤りが存在しない。以上により、エピポーラ線上の探索において、探索範囲を限定した対応点探索が有効であることが確認できる。

奥行き制約により得られた特徴点の追跡結果を利用した形状復元結果を 10 に示す。7 と比較する

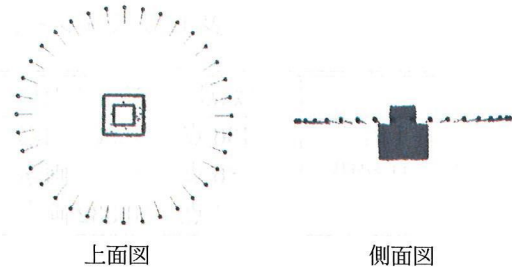


図 10: 奥行き制約を用いた最終的な形状復元結果

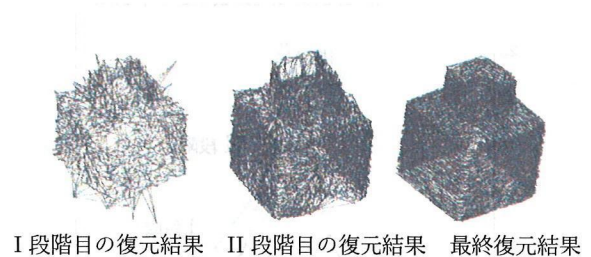


図 11: 各段階の形状復元結果

と、カメラの位置や特徴点の 3 次元座標がより安定して推定されている。

1 および 11 に、アフィン制約を用いた初期形状復元結果 (I 段階) と、奥行き制約を 1 回適用した形状復元結果 (II 段階), 奥行き制約を 2 回適用して得られた最終結果を示す。さらに、復元された 3 次元モデルの精度評価を行うために、3 次元モデルを構成している点群から 1 平面上に存在する点群のみ抜き出し、点群にあてはめた平面の法線ベクトルを求めることで、各面間の平均角度および平面と特徴点との距離誤差の評価を行った結果も 1 および 12 に示した。ここで、物体が存在する仮想的な空間と、形状復元結果の空間では、そのままでは、位置およびスケールが一致していない。これを合わせるために、復元されたカメラ位置と正しいカメラ位置との誤差が最小となる 3 次元アフィン変換を求め変換を行っている。

I 段階目と比較して、II 段階目と最終結果では、特徴点数が大きく増えているにも関わらず、誤差 2 乗和、カメラ位置誤差、直交性の精度が向上しており、高精度に点群が復元できていることが分かる。

表 1: シミュレーション画像による形状復元精度

	特徴点数	誤差 2 乗和	カメラ位置誤差	平均角度	距離誤差範囲
I 段階	1436 点	0.967 画素	0.815 unit	86.3 度	-0.78~1.34 unit
II 段階	9986 点	0.337 画素	0.126 unit	90.0 度	-0.35~0.37 unit
最終結果	11277 点	0.332 画素	0.137 unit	90.0 度	-0.32~0.37 unit

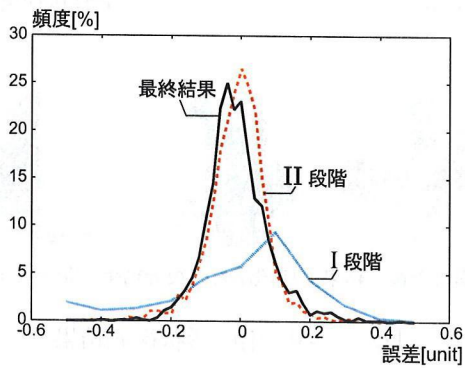


図 12: 平面と特徴点との距離誤差の分布

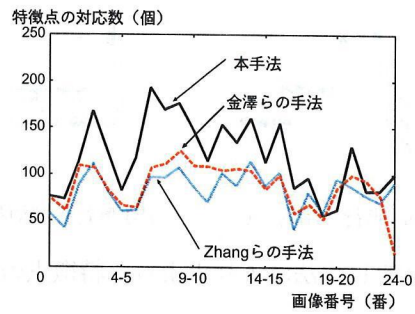


図 13: Zhang, 金澤, および本手法による対応特徴点数の比較

特徴点と平面との位置誤差も、I 段階目では、 $-0.78 \sim 1.34 \text{unit}$  となっていたものが、II 段階目と最終結果では、より狭い範囲となっており、誤差が小さな特徴点の頻度も増えたことから、精度が向上したことが分かる。これは、11 において、I 段階目では物体から外れた点が現れているのに対して、II 段階目以降ではそのような点がなくなっていることから確認できる。

なお、奥行き推定において経験的に決定した値  $\delta = \pm 1.52 \text{unit}$  は、I 段階目における位置誤差範囲よりも大きな値となっており、妥当な値だったと考えられる。

## 5.2 実画像実験

大きさ  $10 \times 8 \times 12 \text{cm}$  の家の模型の全周囲をほぼ等間隔で撮影し、36 枚の画像列を得た。そのうち、11 枚を間引いた 25 枚の画像を用いて実験を行った。画像サイズは  $720 \times 480$  画素とした。

13 に、画像列の各画像における、Zhang らの手

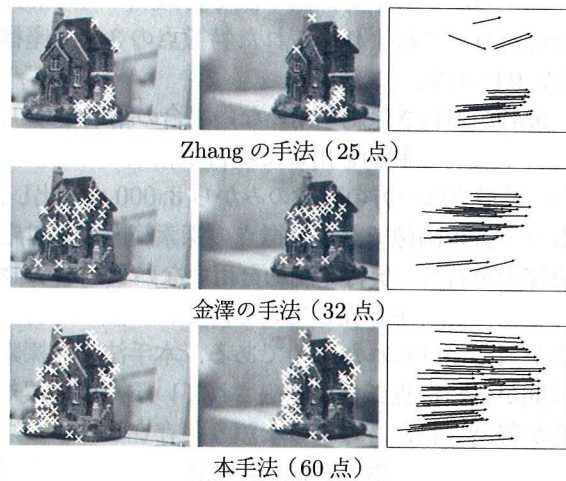


図 14: Zhang, 金澤, および本手法による特徴点の対応付け



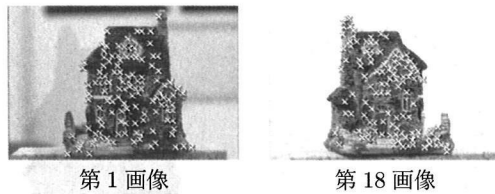


図 15: 特徴点の追跡を行った一例

法 [10]<sup>3</sup>と金澤らの手法 [23]<sup>4</sup> および本手法で得られた対応特徴点の数を示す。なお特徴点は各画像から 300 点検出した。平均対応特徴点数は、Zhang らが 78 点、金澤らが 87 点、本手法が 113 点であり、本手法が最も多くの特徴点を対応付けることができた。

各手法の比較のために、14 に示すように視差が大きく対象物体が画像中心から外れた画像間で特徴点の対応付けを行った。得られた追跡結果は Zhang らが 25 点、金澤らが 32 点、本手法が 60 点であった。14 右は、各手法で得られた対応特徴点のオプティカルフローである。Zhang らの手法では大きな誤対応が存在し、得られる対応特徴点数も少ないことが分かる。Zhang らの手法は、画像間が局所的にアフィン変換により対応付けられると仮定し、得られた特徴点候補に対して、LMedS 推定によりロバストにエピポーラ線を推定している。しかし、14 のような画像では、局所的な制約では正しい特徴点候補が十分得られず、LMedS 推定が有効に機能しないため、このような結果が得られたと考えられる。金澤らの手法では比較的正しい対応がとられているが、得られる対応特徴点数は本手法より少ない。

アフィン制約を適用して対応付けを行った特徴点を 15 に示す（図中の“x”が特徴点を表す）。なお、対応付けには  $23 \times 23$  画素のテンプレートを使用し、アフィン変換で対応する画素を中心に 15 画素以内の領域で SSD を適用して対応特徴点を求めた。その結果、形状を構成する特徴点として 1,294 点が得られた。式 (1) を用いて形状復元

<sup>3</sup><http://www-sop.inria.fr/robotvis/personnel/zzhang/software.html> を利用した。

<sup>4</sup><http://www.img.tutkie.tut.ac.jp/programs/index.html> を利用した。

とカメラ位置・姿勢の推定を行った結果を 16(a) に示す。物体の復元形状とカメラ位置・姿勢は実際の形状や撮影状況にほぼ合致していることを確認した。この結果は、画像数が少ないにも関わらず、カメラ位置・姿勢が対象物体の全周囲を取り囲むように適切に推定されていることを示している。また、3次元上の特徴点群に対し、3角面を与えた結果を 16(b) に示す。特徴点数が少ないため、物体形状は精細でない。しかし、物体の大きな形状を 3角面により表現できている。

次に、奥行き制約を 2 回適用した後の形状復元結果を 17(a) に示す。なお、特徴点は 15,751 点で構成されており、16(b) より 3角面が密に構成されたモデルとなっている。

最後に、各 3角面にテクスチャマッピングを行った結果を 18 に示す。また、他のモデル生成結果を 19 に、生成に関する諸データを 2 に示す。処理は、CPU:Pentium4 2.0GHz、メモリ:1GBytes、OS:Vine linux 2.5 の計算機上で行った。また、奥行き制約の繰り返しは、いずれの例でも 2 回であった。

撮影は、いずれの例でもほぼ等間隔で行った。得られた画像を間引きながら形状復元を行い、画像数が最小となる画像枚数を使用し復元した結果を示した。18 の家の模型では、側面の壁の部分が平面に近いため、11 枚を間引いても全周形状の 3次元モデルが生成できた。一方、19 の石燈籠については、回転体に近い形状をしており、撮影した 35 枚の画像すべてを使わなければ 3次元モデルが生成できなかった。コマインでは、70 枚の画像を撮影したが、結果的には 30 枚の画像から 3次元モデルが生成できた。3次元モデルの生成に必要な画像数は、対象に依存するため理論的な考察は不可能であるが、経験的には、物体の全周囲を 36 枚程度で撮影した画像があれば、本手法により生成が可能であることが多い。

生成された 3次元モデルは、全体として良好といえる。ただし、本研究では物体とほぼ同じ水平面内から撮影を行っているため、特に物体の上面では正しく形状やテクスチャが得られていない。例えば石燈籠の復元結果では、上面は本来尖形を

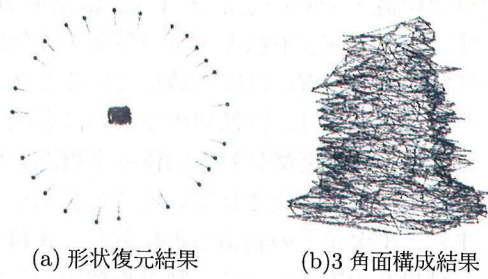


図 16: アフィン制約を用いた初期復元結果

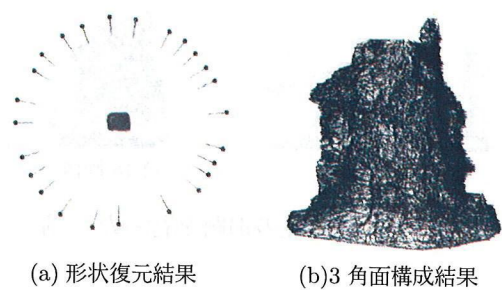


図 17: 奥行き制約を用いた復元結果

表 2: 生成に関する諸データ

画像列	撮影画像	使用画像	特徴点	処理時間
家の模型	36 枚	25 枚	15,751 点	36 分
石燈籠	35 枚	35 枚	4,587 点	34 分
コマイヌ	70 枚	30 枚	10,210 点	18 分

しているが、特徴点が適切にとられず、平面に置き換えられている。

モデル生成の処理時間は、18~36 分程度となっている。このうち多くの部分は、特徴点の対応付けの処理時間である。アフィン制約による特徴点の対応付けでは、連続する全ての 2 画像間で勾配法によりアフィン変換を求めているため、画像数や画像の複雑さによって処理時間が変化するが、1 回あたり 6~17 分程度かかる。奥行き制約による特徴点の対応付けには、3~10 分程度かかり、これを 2 回繰り返している。特徴点の対応付けが与えられた後、非線型最適化により形状復元を行う処理は、特徴点数にもよるが、1 回あたり 3~40 秒程度である。処理時間の短縮は今後の課題であるが、アフィン制約では、解像度を下げた画像でアフィン変換のパラメータを求めた後、もとの画像でパラメータを改良する段階的手法の導入により、処理時間が短縮可能と考えられる。また、奥行き推定では、現在、3 角面とカメラ位置との距離を求める処理に時間がかかっていることから、3 次元的な幾何関係を高速に計算できる GPU 処理などを導入することで、高速化が可能と考えられる。

## 6 おわりに

本論文では、対象物体を撮影した画像列から大局的幾何制約を用いることで、テクスチャ情報を保有した 3 次元モデルを生成する手法を示した。本手法は、まず大局的幾何制約として画像そのもののアフィン変換を利用することで、特徴点追跡を行う。次に、再投影誤差最小化により形状復元とカメラ位置・姿勢の推定を行う。さらに、復元された点群に対し 3 角面を構成し、その奥行きを大局的幾何制約とすることで、より多くの特徴点の追跡を行う。以上の処理を繰り返し行い、得られた 3 角面に対しテクスチャマッピングを行うことで、高精細な 3 次元形状モデルを生成する。

実画像を用いた実験により、本手法は画像間の視差が比較的大きい画像列においても、アフィン変換による大局的幾何制約によって特徴点の追跡が行え、特徴点の 3 次元座標とカメラ位置・姿勢が推定できることを確認した。さらに、3 角面を用いて、その奥行き制約により画像上の探索範囲を限定することで、特徴点の対応付けの誤差を軽減し、安定した 3 次元モデル生成が可能であることを示した。

本研究では、前段階での結果に含まれる誤差を許容する機能を、後段階の処理が持つ手法となっている。このような段階的な方法は、ロバスト性や実用性の観点から重要であり、提案手法により、一貫した手順に従ったモデル生成が可能となる。特に、最終的な形状復元は非線形最適化により求めるため、画像全体で誤差を許容した復元結果を得ることができるという利点がある。





図 18: 3次元モデルの生成



図 19: 他の生成例

ただし、提案手法により任意の画像列から3次元モデルが生成できるわけではない。提案手法で3次元モデルが生成できるかどうかは、画像列の撮影位置、撮影枚数、対象物のテクスチャ、形状、光源状況など多様な要素が影響するため、明確ではない。経験的には、奥行き推定の範囲 $\delta$ や、4.3節の2乗誤差の大きい特徴点の除去において何点除去を行うかというパラメータを調整することで、全周囲を等間隔に36枚程度で撮影した画像列から、3次元モデルが生成できることが多い。ただし、これらのパラメータの設定自体も経験的に行う必要がある、本手法の適用可能範囲を明らかにすることは今後の検討課題である。

また、円筒面への投影を用いた3角面の生成は、奥行き制約として利用するには十分であるが、最終的な3次元モデルとしては、特徴点の接続関係や3角面の形状が視覚的に好ましくない場合がある。この問題を解決することも今後の課題である。

## 参考文献

- [1] T. Oishi and T. Masuda and K. Ikeuchi: "Digital Restoration of the Cultural Heritages," *Proc. VSMM*, pp. 934–941, 2002.
- [2] P. Beardsley, P. Torr and A. Zisserman: "3D Model Acquisition from Extended Image Sequences," *Proc. 4th ECCV'96*, vol.2, pp.683–695, 1996.
- [3] Reinhard Koch, Marc Pollefeys and Luc Van Gool: "Multi Viewpoint Stereo from Uncalibrated Video Sequences," *Proc. ECCV'98*, LNCS, Springer-Verlag, pp.55–71, 1998.
- [4] 佐藤 智和, 神原 誠之, 横矢 直和, 竹村 治雄: "マルチベースラインステレオ法を利用した

- 動画像からの屋外環境の三次元モデル化”, 日本 VR 学論, vol.7, no.2, pp.275–282, 2002.
- [5] M. Lhuillier and L. Quan: “Quasi-Dense Reconstruction from Image Sequence,” *Proc. ECCV’02*, pp.119–139, 2002.
- [6] 中川雅朗, 斎藤英雄, 小沢慎治: “ハンディカメラを用いて撮影された画像列からの EPI の補正に基づく室内環境 3 次元再構築”, 信学論 J84-D-II, pp.266–275, 2002.
- [7] P. Torr and A. Zisserman, “MLESAC: A New Robust Estimator with Application to Estimating Image Geometry,” *Computer Vision and Image Understanding*, Vol. 78(1), pp. 138–156, 2000.
- [8] 佐藤 智和, 神原 誠之, 横矢 直和, 竹村 治雄: “マーカと自然特徴点の追跡による動画像からのカメラ移動パラメータの復元”, 信学論 D-II, J86–D–II, pp. 1431–1440, 2003.
- [9] C. Tomasi and T. Kanade: “Detection and Tracking of Point Features,” *CMU Tech. Rep. CMU-CS-91-132*, 1991.
- [10] Z. Zhang, R. Deriche, O. Faugeras and Q.-T. Luong : “A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry,” *Proc. 3rd Artif. Intell.*, **78**, pp.87–119, 1995
- [11] 金澤靖, 金谷健一: “段階的マッチングによる画像モザイク生成”, 信学論 D-II, J86–D–II, pp. 816–824, 2003.
- [12] M. Okutomi and T. Kanade: “A multiple-baseline stereo,” *IEEE PAMI*, vol. 15, no. 4, pp. 353–363, 1993.
- [13] L. Matthies, T. Kanade and R. Szeliski: “Kalman filter-based algorithms for estimating depth from image sequences,” *IJCV*, vol.3, pp.209–239, 1989.
- [14] G. L. Gimel’farb and R. M. Haralick: “Terrain reconstruction from multiple views,” *Proc. CAIP’97*, LNCS 1296, Springer, pp. 694–701, 1997.
- [15] R. Szeliski and S. B. Kang: “Recovering 3D Shape and Motion from Image Streams using Non-Linear Least Squares,” *Proc. CVPR*, pp.752–753, 1993.
- [16] B. Triggs, P.F. McLauchlan, R.I.Hartley, A. W. Fitzgibbon: “Bundle Adjustment – A Modern Synthesis,” *Vision Algorithms: Theory and Practice*, LNCS 1883, Springer, pp.298–375 2000.
- [17] 右田剛史, 天野 晃, 浅田尚紀: “3 次元形状・運動復元のための高速非線形最適化計算法”, 情処論, 44, No.11, pp.2864–2872, 2003.
- [18] 椋木雅之, 右田剛史, 青山正人, 浅田尚紀: “非線形最適化による建物画像列からの全周形状一括復元のための初期値設定法”, 情処 CVIM 論, 45, Sig13(CVIM10), pp.64–73, 2004.
- [19] A. Fischler, R. C. Bolles: “Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography,” *Comm. ACM*, **24**, pp. 381–395, 1981.
- [20] R. J. Rousseeuw: “Least median of squares regression,” *J. American Stat. Assoc.*, **79**, pp. 871–880, 1984.
- [21] 内藤 洋師, 岡谷 貴之, 出口 光一郎: “画像系列上の幾何学的拘束を用いた 3 次元形状復元”, *MIRU* 2002, pp. II-79–84, 2004.
- [22] 保田和隆, 右田剛史, 青山正人, 椋木雅之, 浅田尚紀: “疎な全周囲画像列からの密な 3 次元形状モデルの生成”, 情処研報, CVIM-138, pp. 73–80, 2003.



- [23] 金澤 靖, 金谷 健一: “大域的な整合性を保証するロバストな画像のマッチング”, 情処 CVIM 論, 44, Sig17(CVIM 8), pp. 70–77, 2003.
- [24] C. Harris and M. Stephens: “A Combined Corner and Edge Detector,” *Proc. Alvey Vision Conf.*, pp. 147–151, 1988.
- [25] Z. Zhang: “Determining The Epipolar Geometry And Its Uncertainty: A Review,” *IJCV*, **27**, pp. 161–195, 1998.

# 3D Model Generation from Image Sequences using Global Geometric Constraint

## Abstract

This paper describes a method for generating a three-dimensional model from an uncalibrated image sequence taken around an object. Our method is based on feature tracking and minimization of re-projection errors. To cope with mis-matchings in the result of feature tracking, we introduce two types of global geometric constraints. The one is “affine constraint” which imposes the positional relationship between pixels on the images. The other is “depth constraint” which imposes the three-dimensional structure of the object. First, we use the affine constraint to reconstruct the object roughly and then we refine the feature tracking and shape reconstruction using the depth constraint. Experimental results show that our method can automatically generate accurate three-dimensional models from real image sequences.

## 1 Introduction

Generating 3D models of real objects is the central concern in computer vision, and many works have been done to recover 3D information from image sequences. When intrinsic and extrinsic camera parameters are known, 3D shape information of the object can be recovered using the geometric relationships between cameras[1],[2]. When the extrinsic camera parameters are unknown, they must be also recovered from the feature points extracted from images[3],[4],[5]. We focus on the latter cases. Most of the methods so far assume that the result of calibration is correct. However, the matching of feature points usually include several errors because of the feature extraction errors, tracking errors, quantization errors of pixel sampling, and so on. As the result, the extrinsic camera parameters and the object shape are poorly reconstructed.

In this paper, we describe a method for generating a three-dimensional model from an image sequence taken around an object, where the extrinsic camera parameters (camera motion) are unknown, while the intrinsic ones are given. Our method is based on feature tracking and minimization of re-projection errors. The method using the minimization of re-projection errors is quite robust for small amount of errors in the feature tracking. However, if the tracking results contain large errors, we cannot reconstruct the object shape. We additionally introduce two types of global geometric constraints. At first, we introduce the “affine constraint” which approximate the transformation between two images by affine transform and restrict the search area for feature tracking within the neighbor of transformed position of the point. Secondly, we

reconstruct the object shape and camera motion from the feature tracking result by minimizing the re-projection errors. Then we eliminate the feature points that have large re-projection errors since such feature points can be wrong matching and decrease the accuracy of the reconstruction. Thirdly, we apply the “depth constraint” which imposes the three-dimensional structure of the reconstructed object. We repeat the second and third stages and refine the feature tracking and shape reconstruction. Finally, we generate a three-dimensional model with texture mapping.

## 2 3D Model Generation

### 2.1 Initial Feature Tracking using Affine Constraint

First, feature points are extracted from each image by Harris operator[6]. We denote the two images of interest as  $I$  and  $I'$ (Fig.1). Each feature point  $P$  extracted from  $I$  is compared with the feature points  $Q$  from  $I'$  using the Sum of Square Differences (SSD).

$$SSD(P, Q) = \sum_{(u,v) \in \Omega, (u',v') \in \Omega'} |I(u, v) - I'(u', v')|^2,$$

where  $\Omega$  and  $\Omega'$  are neighboring region of  $P$  and  $Q$  respectively, and  $I(u, v)$  and  $I'(u', v')$  are pixel values at the  $(u, v)$  and  $(u', v')$  of image  $I$  and  $I'$  respectively.

These points are corresponded in the order of SSD. Here, we introduce the affine constraint. The  $2 \times 3$  affine transform matrix  $A$  is calculated using the global matching of the images.

$$\arg \min_A \sum_{(u,v)} |I(u, v) - I'(u', v')|^2,$$

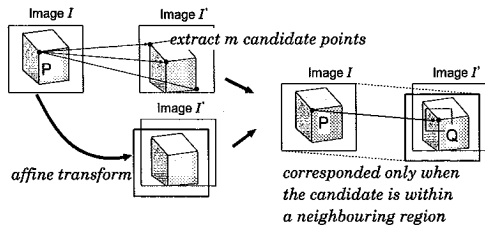


Figure 1: Feature tracking using affine constraint.

$$\text{where } [u' \ v' \ 1]^T = A [u \ v \ 1]^T.$$

We calculate above equation by two steps. In the first step, only the translation parameters are changed all over the image and the optimum parameters are obtained. Then the all 6 parameters of the affine transform is optimized using the gradient descent algorithm where the initial parameters are set as the obtained translation parameters.

A feature point  $Q$  are corresponded to  $P$  only when  $Q$  has within the  $m$ -th smallest SSD and  $Q$  is transformed by affine matrix  $A$  within the  $d$ -pixels away from  $P$ . In the experiments, we use  $m=5$  and  $d=15$ .

## 2.2 Shape Reconstruction

When the result of feature tracking is given, we reconstruct the object shape and camera motion by minimizing the re-projection errors[7][8].

Suppose objects are represented by a set of feature points whose 3D coordinates  $s_p$  ( $p = 1 \cdots P$ ) are defined in the world coordinates, and those feature points are projected onto the  $f$ th image ( $f = 1 \cdots F$ ) (Fig.2). 2D coordinates of the projected  $p$ th point on the  $f$ th image are denoted by  $u_{fp}$ . Note that when the  $p$ th point is not observed from the  $f$ th camera, the corresponding  $u_{fp}$  is undefined in the  $f$ th image.

The pose and position (motion) of the  $f$ th camera are represented by the rotation matrix  $R_f$  and translation vector  $t_f$ . The perspective projection operator  $\mathcal{P}$  is defined by  $\mathcal{P}(x, y, z)^T = \frac{l}{z}(x, y)^T$ , where  $l$  is the focal length of the camera. Using these notations, reconstruction of object shape and camera motion is formulated by minimizing the re-projection error  $E(x)$ , as follows.

$$E(x) = \sum_{(f,p)} |\mathcal{P}[R_f s_p + t_f] - u_{fp}|^2 \quad (1)$$

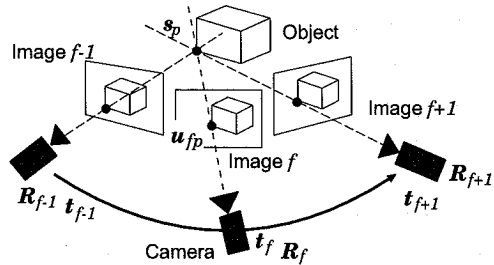


Figure 2: Shape Reconstruction

where  $x$  denotes a composite vector consisting of  $s_p$ ,  $q_f$ ,  $t_f$ , where  $q_f$  is a quaternion representation of the rotation matrix. Since Eq.(1) includes non-linear representation such as perspective projection, the problem is solved by using non-linear optimization method, e.g. Levenberg-Marquardt algorithm, preconditioned conjugate gradient one.

The principle of shape reconstruction by minimizing the re-projection errors is very simple but it gives the optimum result in terms of maximum likelihood estimation for the pixel errors. The method is same as the so called “bundle adjustment[8]”, and widely used for refining and/or integrating the initial and/or partial reconstruction result obtained by other method. We directly apply this method for both of initial reconstruction and refinement steps since it can reconstruct reasonable shape and camera motion even when a small amount of errors exists in the feature tracking result.

On the other hand, the result of feature tracking contains several mis-matchings, which leads to large amount of errors. In general, robust estimation method, such as LMedS estimation or M-estimation[9], is used to cope with this. However, when we use these methods, the convergence of the nonlinear optimization becomes worse and it takes more computational cost. We use the affine constraint for initial feature tracking, which suppresses the outliers. Thus, we directly apply the nonlinear optimization. After the optimization is converged, we calculate the re-projection error for each feature point and delete the point if its re-projection error is large. After that, we again apply the nonlinear optimization to the rest feature points and obtain the object shape and camera motion.

### 2.3 Refinement of Reconstruction using Depth Constraint

Once the object shape and camera motion are reconstructed, we refine them using the depth constraint.

In order to calculate the depth from the camera to the object at each pixel, we make a surface model from the reconstructed feature points. Suppose the feature points are expressed by the 3D coordinates that are centered at the object center and the z-axis is along the object height direction. The feature points are projected to the cylindrical coordinates in order to apply the 2D Delaunay method. When a feature points is expressed  $(x, y, z)$  in 3D coordinates, its projection to the cylindrical coordinates is  $(R, \theta)$ , where

$$R = z \quad \theta = \tan^{-1} \frac{y}{x}.$$

In cylindrical coordinates,  $\theta$  and  $\theta + 2\pi$  denotes the same position. Considering this, feature points projected near the  $\theta = -\pi$  and  $\theta = \pi$  are also projected to the  $\theta + 2\pi$  and  $\theta - 2\pi$  respectively. Then we apply 2D Delaunay method. Finally, we delete the duplicated surfaces and generate the surface model.

Once the surface model is generated, the depth for each pixel can be calculated. The search area of feature tracking is limited within the following area:

$$\begin{bmatrix} u' \\ v' \end{bmatrix} = \mathcal{P} \left[ \mathbf{R}' \hat{\lambda} \begin{pmatrix} u \\ v \\ l \end{pmatrix} + \mathbf{t}' \right] + \begin{bmatrix} 0 \\ \alpha \end{bmatrix} \quad \text{where} \quad \hat{\lambda} \in [\lambda - \delta, \lambda + \delta] \quad (2)$$

$\mathbf{R}'$  and  $\mathbf{t}'$  denotes the camera motion of image  $I'$  relative to image  $I$  and  $l$  is the focus length.  $\lambda$  is the depth of the feature point in the image  $I$  and  $\delta$  is the search area along the depth. Theoretically, the matching feature points lies on the epipolar line. However, the estimated camera motion contains some errors. To cope with this, we also search the feature points along the vertical direction to the image.  $\alpha$  denotes this search area. (Note that this should be vertical to the epipolar line, but for simplicity we employ this setting. Epipolar lines are usually horizontal to the image in our camera motion.)

Feature points are extracted from image  $I$  and the matching is searched from image  $I'$  of the area restricted by the depth constraint using the SSD. This is done to all the successive pair of

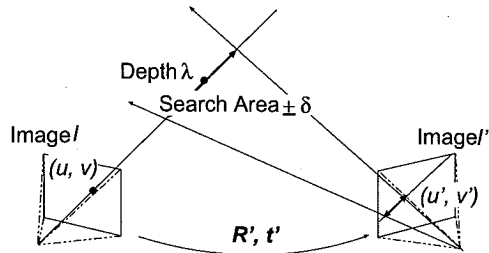


Figure 3: Feature tracking using depth constraint.

images and the new feature tracking result is obtained. The shape reconstruction and the feature tracking are repeated until the number of extracted feature points is converged.

### 2.4 Texture Mapping

In order to generate 3D model, texture is mapped onto the surface model. For each patch of surface, the texture is extracted from the image where all the vertexes of the patch can be observed and the area of the projected patch is the maximum.

## 3 Experiments

In order to evaluate the applicability and characteristics of our method, we apply our method to a real image sequence. The object is a model house with  $10 \times 8 \times 12cm$  large. We take 25 images around the object, each of which has  $720 \times 480$  pixels.

Fig.4 shows the number of tracked feature points for each image. For comparison, we applied Zhang's method[10] and Kanazawa's method[11] to the same image sequence. The mean number of tracked feature points was 113 points by our method (with affine constraint) and 78 points by Zhang's method and 87 points by Kanazawa's method. Fig.5 is another example for demonstrating the characteristics of our method. Zhang's method can track smaller number of feature points and the points are not uniformly distributed on the object. The optical flow obtained by our method shows that the result of our method does not contain obvious outliers. Although Kanazawa's method does not contain obvious outliers as well, it apt to track smaller number of feature points.

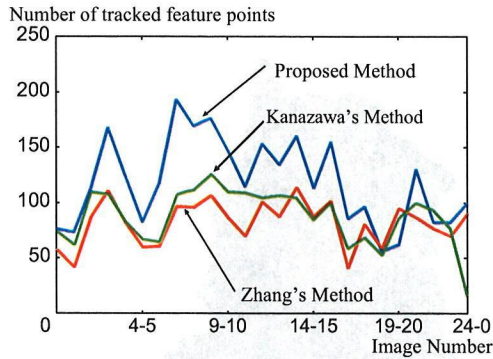


Figure 4: Number of tracking points by Zhang's method, Kanazawa's one and the proposed one(affine constraint).

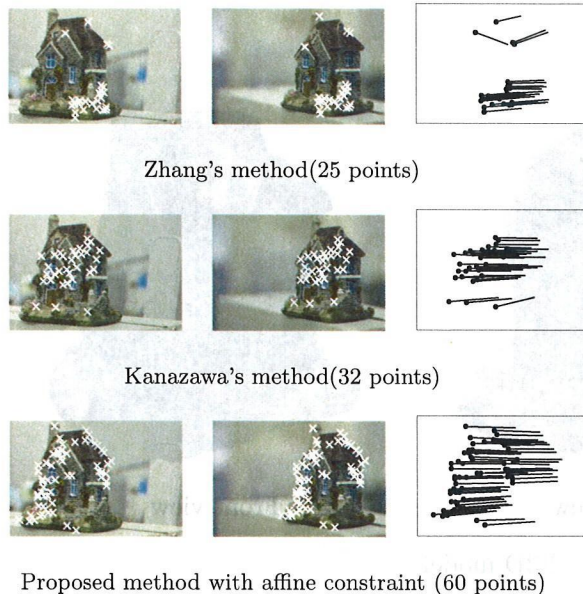


Figure 5: Feature points and optical flow.

Fig.6(a) shows the initial reconstruction result using affine constraint. The result consists of 1,294 unique feature points. The initial result holds the substantial object shape and camera motion although it contains some errors and the generated model is rough.

Fig.6(b) shows the final reconstruction result using depth constraint. The result consists of 15,751 unique feature points. Fig.7 shows the generated 3D model with texture mapping. Fig.8 shows the other results of generated 3D models. We can automatically generate these 3D models from the given image sequences.

## 4 Conclusion

We have proposed a stepwise refinement approach for 3D shape reconstruction. We employed the minimization of re-projection errors for shape reconstruction method and introduced two types of global constraints. We could stably obtain initial reconstruction using affine constraint and refine it using depth constraint. Our procedure has applied to several kinds of real image sequences and we can obtain fine 3D models. We think our method is applicable to relatively wide variety of real image sequences.

At present, we assume that the camera motion is circular around the object. The texture on the top of the object cannot be obtained under this camera motion. It is remained as future works to use the images from every direction for reconstruction and 3D model generation.

## References

- [1] L. Matthies, T. Kanade and R. Szeliski: "Kalman filter-based algorithms for estimating depth from image sequences," *International Journal of Computer Vision*, vol. 3, pp. 209-239, 1989.
- [2] G. L. Gimel'farb and R. M. Haralick: "Terrain reconstruction from multiple views," *Proc. 7th Int. Conf. on Computer Analysis of Images and Patterns (CAIP'97)*, LNCS 1296, Springer, pp. 694-701, 1997.
- [3] P. Beardsley, P. Torr and A. Zisserman: "3D Model Acquisition from Extended Image Sequences," *Proc. ECCV'96*, Cambridge, UK., vol. 2, pp. 683-695, 1996.
- [4] R. Koch, M. Pollefeys and L. V. Gool: "Multi Viewpoint Stereo from Uncalibrated Video Sequences," *Proc. ECCV'98*, LNCS, Springer, pp. 55-71, 1998.
- [5] R. Hartley and A. Zisserman, : "Multiple View Geometry in Computer Vision," *Cambridge University Press*, Cambridge, U.K, 2000.
- [6] C. Harris and M. Stephens: "A Combined Corner and Edge Detector," *Proc. Alvey Vision Conf.*, pp. 147-151, 1988.
- [7] R. Szeliski and S. B. Kang: "Recovering 3D Shape and Motion from Image Streams using Non-Linear Least Squares," *CVPR*, pp.752-753, (1993).



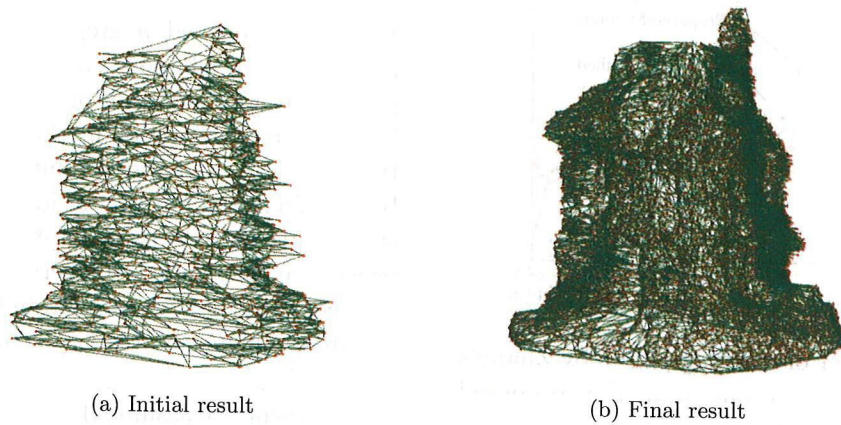


Figure 6: Reconstruction result



Figure 7: Generated 3D model

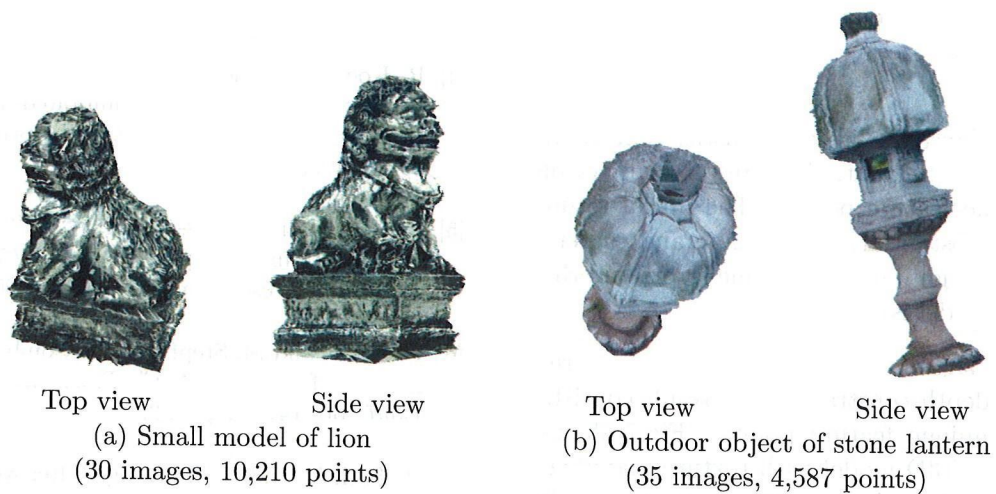


Figure 8: Other results



- [8] B. Triggs, P.F. McLauchlan, R.I.Hartley, A. W. Fitzgibbon: "Bundle Adjustment – A Modern Synthesis," *Vision Algorithms: Theory and Practice*, LNCS 1883, Springer, pp.298-375 2000.
- [9] Z. Zhang: "Determining The Epipolar Geometry And Its Uncertainty: A Review," *IJCV*, **27**, pp. 161–195, 1998.
- [10] Z. Zhang, R. Deriche, O. Faugeras and Q.-T. Luong : "A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry," *Proc. 3rd Artif. Intell.*, **78**, pp. 87–119, 1995
- [11] Y. Kanazawa and K. Kanatani: "Robust Image Matching Preserving Global Consistency", *Proc. 6th ACCV*, pp.1128–1133, 2004.

# 奥行き制約による対応点探索を用いた 画像列からの3次元モデル生成

## 1 はじめに

実物体の3次元モデル化は、仮想空間におけるリアルな物体表現に利用できるだけでなく、文化財のデジタルアーカイブなどさまざまな目的に用いることができる技術である [1]。そのため、3次元モデル生成にはさまざまな手法が提案されている。その一つにレーザレンジセンサなど専用の計測装置により高精細な3次元モデルを生成する手法がある。しかし、この測定機器は高価だけでなく、対象物体の大きさが限られていたり、対象物の色、反射率、透過率が測定に影響するなどの制約も多い。そのため、従来から手軽に3次元モデルを生成する手法の研究が行われている。その一つに多視点で撮影された画像列から3次元モデルを生成する手法がある。画像列から形状復元を行う手法では、特殊な機器が不要であり、対象物体の大きさに制限されにくいといった利点があるため、多くの研究が行われている [2, 3, 4]。

これらの手法では複数の画像を撮影し、画像間での点を対応付けする必要がある。対応点が正確に求められれば、透視投影のモデルに基づき各点の3次元座標が計算可能である。しかし、点の対応付けは見え方の変化や隠蔽、フレームアウトなどにより誤対応を生じるため難しい問題である。そこで、Zhang[5]らや金澤・金谷 [6] は、特徴点のランダム選択と局所対応の投票などを行い、特徴点の対応付けを行う手法を提案している。また、カメラ位置・姿勢が既知ならば、マルチベースラインステレオ法 [7] や多視点解析 [8] などにより、点群からなる部分モデルを生成できる。ただし、視差が大きい画像列では、カメラ間距離（ベースライン）が大きくなるため、マルチベースラインステレオ法は利用できない。また、特徴点の誤対応による累積的な推定誤差が大きくなり、部分モデルの統合は容易ではない。このように個々の技術については多数の研究が行われている。一方、保田ら [9] は前提となる条件の誤差を許容しつつ、新しい制約を追加しながら点の対応付けと3次元形状を逐次的に改善することで物体

の全周囲を撮影した画像列から高精細な3次元モデルを生成する手法を提案している。

保田らの手法を用いると、物体の全周囲を撮影した20から40枚程度の画像列より高精細な3次元モデルを生成できる。しかし、面生成の特性により、地面などの水平面を含む対象や、凹形状の対象については3次元モデルの生成が良好に行えない。例えば、上空から複雑な形状をした建物を撮影した画像列からは、地面を含む3次元モデルの生成を行うことができなかった。

本研究では上空から撮影された画像列からも3次元モデルを生成できるよう、保田らの手法に新たな手法を追加し、改良する。以後、単に従来手法と述べている場合は保田らの手法のこととする。本手法では画像列から3次元モデルを生成するまでを次の3段階に分ける。

1段階目では従来手法と同様、アフィン制約を用いて特徴点群による疎な形状復元を行う。2段階目では1段階目で取得した特徴点の近傍において、新たな特徴点の奥行きを推定することにより特徴点の対応付けを行うことで特徴点群による密な形状復元を行う。なお、2段階目は繰り返しを行い、より密な形状復元を目指す。最後に3段階目ではマーチンキューブ法を用いて面を生成し、面の法線ベクトルを利用してテクスチャの選択、付加を行うことで3次元モデル生成を行う。

以下、2節では座標系などの諸定義と従来研究の流れについてまとめる。続いて3節では上空撮影画像列の特徴・問題点と解決するための新たな手法の概要、4節では本手法全体の流れと新たな手法の詳細について述べる。5節では各段階における有効性の評価および同一の画像列を用いた従来手法との比較を行う。最後に第6章では本論文のまとめを示す。

## 2 画像列からの形状復元

### 2.1 座標系の設定

図1に示すように、現実空間をモデル化した3次元空間を考える。そこに固定された座標系を**世界座標系**とする。次に、その3次元空間を自由に動き回るカメラを考える。ここで、カメラの焦点位置  $C$  を原点とした局所座標系を**カメラ座標系**とする。

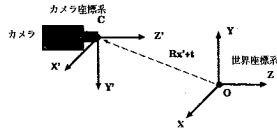


図1: カメラ座標系と世界座標系の関係

ある点のカメラ座標系における座標  $\boldsymbol{x} = (x, y, z)^T$  と、世界座標系における座標  $\boldsymbol{x}' = (x', y', z')^T$  との関係は次のように表される。

$$\boldsymbol{x} = \boldsymbol{R}\boldsymbol{x}' + \boldsymbol{t} \quad (1)$$

ここで、 $\boldsymbol{t}$ は並進ベクトル、 $\boldsymbol{R}$ は回転行列である。これらは世界座標系におけるカメラの位置と姿勢を表すパラメータである。以後これらのパラメータをカメラ位置・姿勢と呼ぶ。

次に、物体と、カメラで撮影された物体の投影される画像平面の関係を図2に示す。カメラの焦点位置  $C$  から画像平面  $I$  までの距離  $f$  を**焦点距離**と呼び、カメラの焦点位置  $C$  を通り画像平面  $I$  に垂直な線を**光軸**、光軸と画像平面  $I$  との交点  $c$  を**画像中心**と呼ぶ。

また画像中心  $c$  を原点とし、 $u$  軸と  $v$  軸をそれぞれ画像の画素の配列軸に合わせた2次元座標系を**画像座標系**と呼ぶ。ここで画像に対して右向きに  $u$  軸、下向きに  $v$  軸をとる。カメラ座標系は、焦点位置  $C$  を原点に光軸を  $Z$  軸とし、 $Z$  軸と直行する  $X$  軸と  $Y$  軸はそれぞれ画像座標系の  $u$  軸と  $v$  軸に一致する。

3次元空間中の点  $M$  が、画像平面  $I$  上で  $m$  に投影されたとすると、点  $M$  とカメラの焦点位置  $C$ 、画像平面  $I$  上の像  $m$  は一直線上にある。この関係を透視投影 (Perspective Projection) という。前述の座標系を用いると、3次元空間内の点  $M$  のカメラ座標系での座標  $(x, y, z)^T$  とその2次元像  $m$  の

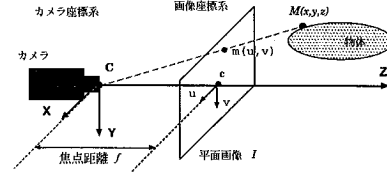


図2: 新たな特徴点候補点の推定座標

画像座標  $(u, v)^T$  との間に次式のような関係が成り立つ。

$$\begin{bmatrix} u \\ v \end{bmatrix} := \mathcal{P} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad \text{ここで } \mathcal{P} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \frac{f}{z} \begin{bmatrix} x \\ y \end{bmatrix} \quad (2)$$

ここで  $\mathcal{P}$  は透視投影を表す演算子である。

点  $M$  を世界座標系での座標  $(x', y', z')^T$  で表すと、2次元像  $m$  は世界座標系からカメラ座標系への変換を含む次式で表される。

$$\begin{bmatrix} u \\ v \end{bmatrix} = \mathcal{P} \left[ \boldsymbol{R} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} + \boldsymbol{t} \right] \quad (3)$$

ここで、 $\boldsymbol{R}, \boldsymbol{t}$  はカメラ位置・姿勢である。

### 2.2 大局的幾何制約を用いた3次元モデル生成

従来研究である大局的幾何制約を用いた3次元モデル生成 [9] の流れは以下の通りである。

#### 1. 画像列の撮影

対象物の回りを回りながら撮影を行う。この時、撮影の間隔をあまり広げ過ぎないようにする。

#### 2. 特徴点の検出

対象を撮影した画像列を入力とし、各画像から Harris の特徴点検出手法 [10] を用いて、特徴点を数百点検出する。

#### 3. アフィン制約による特徴点の対応付け

各画像から検出された特徴点に対して、画像全体をアフィン変換で近似を行うという大局

的幾何制約を用いて、対応付けを行う。これをアフィン制約とよぶ。

#### 4. 形状復元とカメラ位置・姿勢の推定

特徴点の対応付け結果に対して、仮の形状復元を行う。仮の復元結果を用いて各特徴点の再投影誤差を評価し、誤差の大きい特徴点を除去する。残った特徴点の3次元座標を元にしてカメラ位置・姿勢を再度推定する。

#### 5. 2次元ドローネー法を用いた3角面生成

3次元上の特徴点座標を、円筒面に投影した点群に対して、2次元ドローネー法を適用し、3角面を生成する。

#### 6. 3角面の奥行き制約による対応点の対応付け

特徴点で構成された3角面と、カメラ位置・姿勢を用いることで、エピポーラ線上の対応点探索範囲を限定し、より密な特徴点の対応点の対応付けを行う。4から6までを特徴点数が安定するまで繰り返す。

#### 7. 面生成

最終的な3次元点群に対して、5と同様の処理により3角面を生成する。

#### 8. テクスチャ画像選択

得られた3角面の3次元座標と隣接関係を用いて、全周囲画像列より各3角面に適切なテクスチャ画像を選択する。

### 2.3 上空撮影画像列の利用

従来手法で対象としていた画像は地上から撮影された画像列、すなわち対象物の側面を撮影した画像列である。そのため、地面はほとんど含まれておらず、屋根も撮影不可能である。しかし、図3に示すように実在する建造物の場合、地面や屋根がないと、3次元モデルとしては不完全な印象を与えやすい。

そこで、上空から撮影された画像列を使用することを考える。これにより、地上からでは撮影できなかった地面や屋根を含んだ画像列が得られ、地面と屋根のついた3次元モデルの生成が行える。

しかし、上空から撮影を行った画像列と、地上で撮影を行った画像列には地面や建物の屋根以外にも大きな違いがある。それは図4に示すような大き

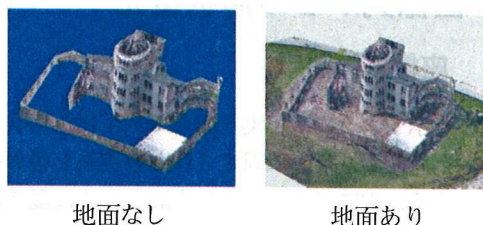


図3: 地面の有無による視覚的变化

な外枠の中に物体があるような凹形状の場合に発生する。

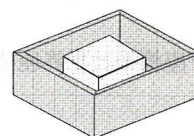


図4: 外枠の中に物体がある形状

図4に示す形状を地面から撮影する場合、まわりの側面のみを撮影するため、中の空間や直方体は撮影されないため、復元できない。しかし、上空から撮影した場合、内側の壁や直方体も撮影されるため、復元の対象となる。

従来手法では3次元の点群を図5のように円筒面に投影し、2次元ドローネー法により面を生成している。よって、図4のような形状に面を生成すると図6のように内側の物体と外枠の間にも面が生成され、本来の形状とは異なる形状となる。

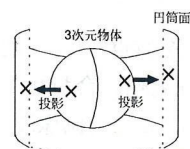


図5: 従来手法の問題点

従来手法では面の生成を、奥行き制約と3次元モデルへのテクスチャ付加において利用していた。このうち、奥行き制約に関しては生成された面群に誤りが発生すると図7のような誤った奥行きが推定されるため、特徴点の誤対応が発生する。特徴点の誤対応が存在すると正確な形状復元は不可能である。

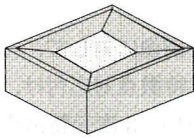


図 6: 復元された異なる形状

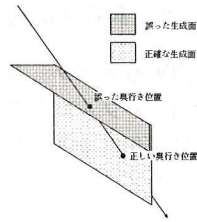


図 7: 誤った奥行き

この奥行き制約を用いた形状復元を行うことで多数の特徴点を取得しているため、誤対応点の発生は3次元モデルの細かな形状の精度に大きな影響を与える。よって、手法の変更が必要である。そこで本研究では既知の特徴点を球として、その近傍から特徴点候補の奥行きを推定する(図8)。

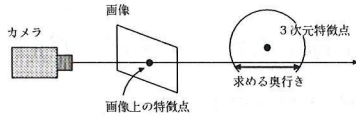


図 8: 本研究の奥行き推定法

また、3次元モデルへのテクスチャ付加においては、本研究では面生成手法としてマーチンキューブ法を用いる。マーチンキューブ法を用いると隣接しない点間に面を張ることはなく、凹形状の復元に適している。しかし、マーチンキューブ法は粗い点群による形状復元には使用できず、仮に使用しても十分な面を生成することはできない。よって、密な復元が行えた最終段階のみマーチンキューブ法を使用する。これらの手法について次章で説明する。

### 3 3次元モデルの生成処理

#### 3.1 生成処理の流れ

本手法による上空撮影画像列より3次元モデルを生成するまでの手法について示す。本手法の処理の流れと処理に対する入力データ、出力データの関係を図9に示し、次節以降で従来手法より変更した箇所の詳細を示す。なお、図9において、四角で囲まれた内容が処理を示し、楕円に囲まれた内容が入力データと出力データを示している。

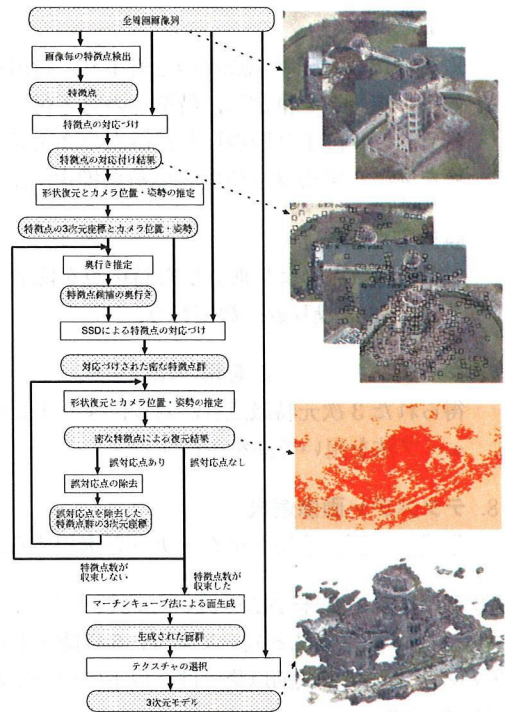


図 9: 3次元モデル生成までの流れ

#### 1. 画像列の撮影

対象物の回りを回りながら撮影を行う。この時、撮影の間隔をあまり広げ過ぎないようにする。

#### 2. 特徴点の検出

対象を撮影した画像列を入力とし、各画像から Harris の特徴点検出手法 [10] を用いて、特徴点を数百点検出する。

#### 3. アフィン制約による特徴点の対応付け

各画像から検出された特徴点に対して、画像全体をアフィン変換で近似を行うという大局的幾何制約を用いて、対応付けを行う。

#### 4. 形状復元とカメラ位置・姿勢の推定

特徴点の対応付け結果に対して、仮の形状復元を行う。仮の復元結果を用いて各特徴点の再投影誤差を評価し、誤差の大きい特徴点を除去する。残った特徴点の3次元座標を元にしてカメラ位置・姿勢を再度推定する。



### 5. 特徴点の近傍による奥行き制約を用いた形状復元

取得した3次元特徴点群の近傍より、特徴点候補の奥行きを推定し、特徴点の対応付けを行う。これにより取得した特徴点の3次元座標を元にしてカメラ位置・姿勢を推定する。

### 6. 誤対応点除去

3次元特徴点群より孤立した特徴点を除去する。5, 6は繰り返し数回行う。

### 7. マーチンキューブ法を用いた面生成

得られた3次元特徴点群に対し、マーチンキューブ法を用いて面を生成する。

### 8. テクスチャ画像選択

生成した面に適切なテクスチャを選択する。

2.3章で示した従来手法の流れと比較すると、1から4までは従来手法と同一であり、5以降を変更している。よって、次節以降では5以下について順に詳細を示す。

## 3.2 特徴点の近傍による奥行き制約を用いた形状復元

### 3.2.1 特徴点の対応付けと形状復元

本手法の奥行き推定法は、既知の3次元特徴点と、近傍にあたる特徴点の奥行きは非常に近いと推測できることを利用したものである。概要は以下の通りである。

1. 3次元上の各特徴点がある一定の半径を持つ球とみなす。
2. 画像から特徴点を検出し、その3次元上の頂点の奥行きを球と交わる線分として求める。
3. 求めた奥行きの周辺を探索範囲とし、対応付けを行う画像上に投影する。
4. 投影された領域から特徴点を対応付ける。
5. 特徴点の対応付け結果より形状復元を行う。
6. これを特徴点数が安定するまで繰り返す。

以下に1~4の詳細を示す。

### 3.2.2 奥行き制約

**入力するデータ** 与えられた画像に対して、特徴点検出、アフィン変換を用いた特徴点对応付け、非線形最適化による形状復元を行い、3次元特徴点を求めておく。

**第1段階** ある画像  $I$  について、3次元特徴点をその画像を撮影したカメラの焦点位置  $C$  を原点とするカメラ座標系に変換する。

**第2段階** Harrisの検出法[10]により画像から特徴点候補を求める。

求めた特徴点候補点  $m(u, v)$  について、図10に示すように、原点と点  $m$  を通る直線と3次元特徴点  $N(x, y, z)$  を中心とする球が交わる特徴点  $N$  を次式の値  $A$  より求める。

$$A = (ux + vy + fz)^2 - (u^2 + v^2 + f^2)(x^2 + y^2 + z^2 - r^2) \quad (4)$$

ここで、値  $A$  が0以上の時、直線と球は交わることを示し、値  $r$  は球の半径であり、値  $f$  はカメラの焦点距離である。

次に、特徴点  $N$  を中心とする球と、原点と特徴点候補点を通る直線との交点  $M1(x', y', z')$ ,  $M2(x'', y'', z'')$  を次式により求める。

$$x' = \frac{u}{f}z' \quad (5)$$

$$x'' = \frac{u}{f}z'' \quad (6)$$

$$y' = \frac{v}{f}z' \quad (7)$$

$$y'' = \frac{v}{f}z'' \quad (8)$$

$$z' = \frac{f(ux + vy + fz) + f\sqrt{A}}{u^2 + v^2 + f^2} \quad (9)$$

$$z'' = \frac{f(ux + vy + fz) - f\sqrt{A}}{u^2 + v^2 + f^2} \quad (10)$$

**第3段階** 交点  $M1, M2$  はカメラ座標系の値なので他のカメラ座標系に投影できるようにするため、次式により世界座標系における推定座標点  $M1'(x_1, y_1, z_1)$ ,  $M2'(x_2, y_2, z_2)$  に変換する。

$$\begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} = \mathbf{R}^{-1} \left( \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} - \mathbf{t} \right) \quad (11)$$

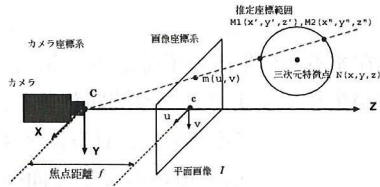


図 10: 新たな特徴点候補点の推定座標

$$\begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix} = R^{-1} \left( \begin{pmatrix} x'' \\ y'' \\ z'' \end{pmatrix} - t \right) \quad (12)$$

ここで  $R^{-1}$  は回転行列  $R$  の逆行列である。

連続した画像  $I-1, I+1$  のカメラ座標系を求め、これにより、図 11 に示すように画像  $I$  の特徴点候補の推定座標点  $M1', M2'$  を画像  $I-1, I+1$  上に投影する。

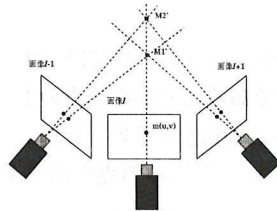


図 11: 特徴点の対応付け

**第 4 段階** 投影した 2 点を結ぶ線分を対角線とする長方形領域を奥行き推定領域とする。画像  $I-1, I+1$  それぞれの奥行き推定領域内の点について SSD を用いて類似度を計算し、類似度がしきい値以下の場合、最適な対応点を選択する。これにより、3 画像での特徴点の対応付けを行うことになる。

しかしこの方法では、以下の図 12 のように画像  $I, I+1$  の特徴点候補から特徴点の対応付けを行った時に、画像  $I+1$  では同一の点が隣の画像  $I$  では異なる点となることがある。そこでそれぞれの点の輝度値を SSD により比較し、より値の小さな点を残し、残りの点を削除する。

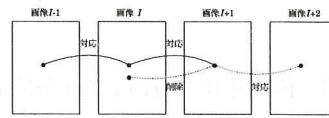


図 12: 重複した特徴点の除去

### 3.3 3次元特徴点の距離による特徴点の除去

以上により取得した特徴点を用いて 3 次元特徴点群による形状復元を行う。図 13 は復元を行った結果の一例である。

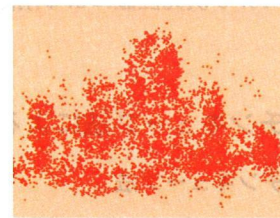


図 13: 3次元特徴点群による形状復元結果

図 13 より、孤立した 3 次元特徴点があることが分かる。この孤立した 3 次元特徴点を対応付けを行った画像上で示すと図 14 の中心の点となり、誤対応ではあるが正しい対応位置に非常に近い位置に点がある。これより、従来研究で誤対応の特徴点除去に使用していたロバスト推定では除去が難しい。

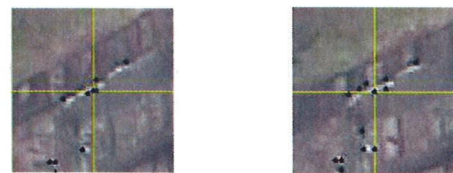


図 14: 誤対応点

誤対応点は画像上で比較しても誤差が良く分からないが、3 次元特徴点群のなかでは孤立している。そこで 3 次元特徴点間の距離から特徴点の除去を行った。以下にその手法を示す。

1. ある特徴点から距離がしきい値  $M$  以下の特徴点数を数える。



2. その特徴点数がしきい値  $N$  以下の点を除去する。

この結果を以下の図 15 に示す。(a) が除去前、(b) が除去後である。これより、孤立した点の除去が行えた。しかし、この手法は孤立した特徴点であれば正確な特徴点も除去してしまうため、改良の余地がある。

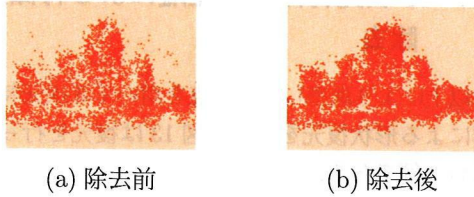


図 15: 3次元位置による削除

### 3.4 マーチンキューブ法による面生成

#### 3.4.1 マーチンキューブ法

面を生成する手法として、マーチンキューブ法を使用する [12]。マーチンキューブ法とはボリュウムレンダリング法の一つで、3次元行列内に蓄積されたデータから、一定の濃度の表面を多角形で近似する手法である。

8個のボクセルを頂点とする立方体に対してボクセル値のパターンを用いて等値面を推定し面を生成する。まず、生成する等値面のしきい値と各ボクセル値を比較しその大小により各ボクセルを2種類に分類する。次に各立方体の頂点のボクセル値の組み合わせパターンから等値面の3角パッチのパターンを求める。このとき立方体の8点の組み合わせパターンは $2^8$ の256通りである。しかし、回転・対象を除く場合、図 16 に示すように15パターンで表すことができる。マーチンキューブ法を用いること

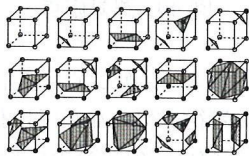


図 16: マーチンキューブ法

により、ボクセルから容易に等値面を推定すること

ができる。しかし、隣接する立方体との組み合わせにより、図 17 に示すような場合、マーチンキューブ法に存在する面の生成の不確定性から、面が生成されず穴が生じてしまうという問題がある。

この問題点を改善するために、立方体を4面体に分割する4面体分割法を用いることで面の生成の際の不確定性を取り除く [13]。4面体分割法は、4面体の頂点のボクセルにマーチンキューブ法と同等の方法で等値面を生成する。マーチンキューブ法に不確定性が生じる原因として、隣接する立方体との共有面が4角形であることが挙げられる。そこで、4面体に分割し、隣接する4面体との共有面を3角形にすることで、この問題を改善する (図 18)。

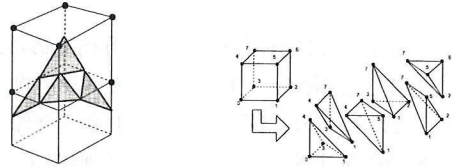


図 17: マーチンキューブ 図 18: 6面体を4面体へ法の問題点

また、マーチンキューブ法では立方体の頂点のボクセルは8個であることから頂点のボクセル値のパターンは $2^8$ の256通りであったが、4面体分割法を用いることで4面体の頂点のボクセルは4個になるためボクセル値のパターンは $2^4$ の16通りであり、回転・対象を除くと図 19 に示す2通りで表せる。

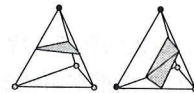


図 19: 4面体分割法において生成される面

#### 3.4.2 点群からの3次元モデル生成

マーチンキューブ法はボクセルに与えられたスカラー値に基づき面を生成する。よって3次元特徴点データから面を生成するためには3次元特徴点データをボクセル空間に投影し、ボクセルにスカラー値を与える必要がある。今回は投影する手法として、以下の手法を用いた。

● 3次元座標からボクセル空間への変換

3次元特徴点全てを含む直方体空間を  $x, y, z$  座標の最大値と最小値で定まる  $I_x \times I_y \times I_z$  に分割する. さらに, 立方体空間の周囲にも面が生成されるはずなので,  $x, y, z$  の周辺に3ボクセルを加え, ボクセル空間は  $(I_x + 6) \times (I_y + 6) \times (I_z + 6)$  とした. ここで, ボクセルそれぞれのスカラー値を  $voxel(i, j, k)$  とする.  $i, j, k$  はそれぞれ分割したボクセルの位置である.

● 特徴点

3次元特徴点が対応するボクセルに1.0を加える. 以後, 値を加えることを投票と呼ぶ. さらに, 値を投票したボクセルの頂点から上下左右前後の6つのボクセルの頂点に0.5, 斜めに隣接する8つのボクセルに0.2を投票する.

以上をすべての3次元特徴点について行うことで, ボクセルにスカラー値を与えることができる. さらにしきい値  $S$  を設定し, スカラー値としきい値の大小よりマーチンキューブ法を適用することで面を生成する.

次にマーチンキューブ法により取得された面データに対する, テクスチャの張り付けについて説明する. 本手法では面の法線ベクトルとカメラ位置から面へのベクトルの内積の値によりテクスチャを選択する. 以下に手法を示す.

1. 注目している面を中心とした各面の法線ベクトルを図20に示す重み付けにより加算し, その面の法線ベクトルとする.
2. 1をすべての面について行う.
3. カメラ位置から面へのベクトルと, 面の法線ベクトルの内積の値を求める.
4. 3をすべてのカメラ位置について行う.
5. 各面について, 内積が最小となったカメラ位置に対応する画像よりテクスチャを取得する.

以上により生成された3次元モデルを以下の図21に示す.

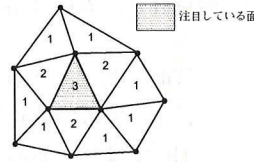


図 20: 重み付け



図 21: テクスチャ付加結果

## 4 実験

### 4.1 奥行き制約による特徴点对応付け

特徴点の近傍による奥行き制約を用いて取得した新たな特徴点を入力として繰り返し特徴点对応付けする実験を行った. 奥行き推定手法で使用している点を中心とした球の半径は  $5/f$ , 対応付けにおけるSSDの小領域は  $5 \times 5$ , しきい値は10000に固定した. ここで  $f$  とは焦点距離であり半径  $5/f$  の球は, 点群の中心付近に球を置いた時に画像上では半径5画素の円となる.

対象物体は広島原爆ドームである. 画像は原爆ドーム上空を旋回してビデオにより撮影した画像列2650枚より, 40枚刻みで選別した66枚 ( $480 \times 320$ 画素) を使用した. 図22に使用した画像の一部を示す.

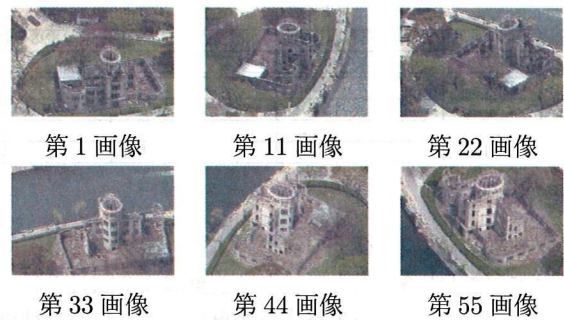


図 22: 使用した画像列:原爆ドーム

取得した特徴点群による復元結果で特徴的な一部分を図23,24に示す. なお, 図中の(a)はU字型になった壁を上から, 図中の(b)はドーム部分を横から見た様子である. また, それぞれの特徴点数を表1と図25に示す. ただし, 初期段階はロバスト推定により, 250点の削除を8回行ったものであり,



以降は取得した特徴点群から3次元位置に基づき $(5/f)$ の距離に2点以下の特徴点群を除去した結果である。

図より、2回目までは形状が鮮明であり、特徴点も増加しているが、3回目からは(a)では特徴点に変形して形状が不鮮明となり、(b)ではドーム部分の形状が変形していることがわかる。表1より、3回目までは特徴点数が増加しているが、4回目は減少している。また、削除した特徴点数は回数を重ねる毎に増加する傾向がある。

これより、何度も繰り返すことで特徴点が増えていくわけではなく、削除数が増加したことから形状が拡散する傾向が読みとれる。よって取得した特徴点群を入力として奥行き推定手法を繰り返すことは、2回程度なら効果があるが、それ以上行くと形状が変形したり、特徴点数が減少するなどの問題が発生することがわかった。

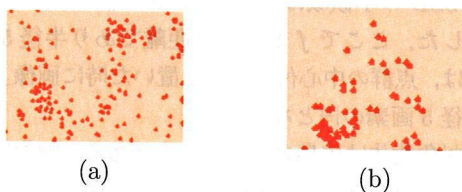


図 23: アフィン制約による初期段階での復元結果

表 1: 各段階の特徴点数

	取得した特徴点数	削除した特徴点数	削除後の特徴点数
初期復元	3538	2000	1538
1回目	10044	728	9316
2回目	15268	728	14540
3回目	15553	1517	14036
4回目	14681	1900	12781

次に、球の半径 $r$ の変更による、形状の変形について検証する。入力データは前実験の2回目のデータを使用し、球の半径 $r$ を変更して復元した。結果を図26、27と表2に示す。なお、図中の(a)はU字型になった壁を上から、図中の(b)はドーム部分を横から見た様子であり、削除の条件は前実験と同一である。

復元形状を調べると、半径が $1/f$ では壁はあまり変形せず、前実験で起きたドーム部分の形状の変形もほとんど見られない。一方、特徴点数は半径が $2/f$ の時最大となることが分かる。

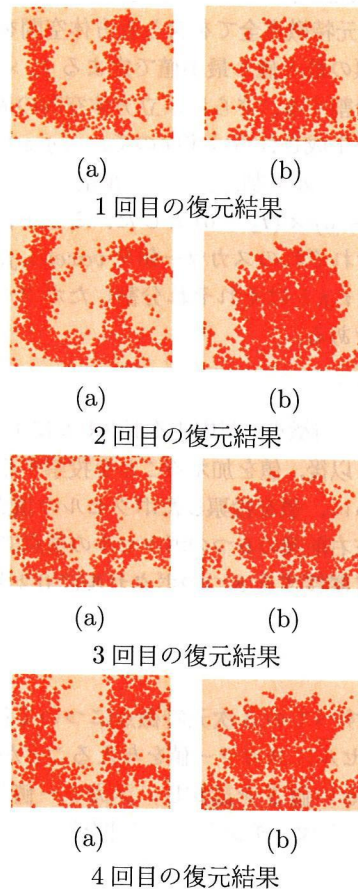


図 24: 繰り返しによる形状の変化

半径 $r$ を小さくすると特徴点数は減少すると思っていたが、 $r = 2/f$ までは増加する結果となった。これは $r$ が大きい時に奥行き推定が誤っていた特徴点候補が正しく奥行きが推定され、対応付け出来たためだと考えられる。

しかし、削除後の特徴点数はすべて減少している。半径を変更して手順を繰り返しても形状が不安定になり、特徴点数も減ることから半径の変更は復元形状の改善には有効ではないと考えられる。

表 2: 半径 $r$ と特徴点数

	取得した特徴点数	削除した特徴点数	削除後の特徴点数
初期復元	15268	728	14540
半径 $4/f$	14909	1912	12997
半径 $3/f$	15361	1725	13636
半径 $2/f$	15929	1541	14388
半径 $1/f$	15753	1469	14284

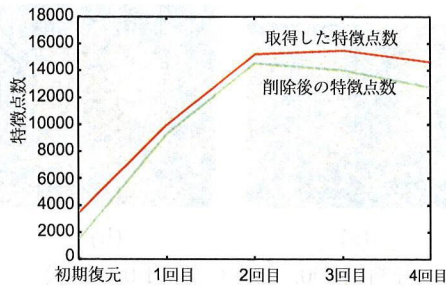


図 25: 各段階の特徴点数

## 4.2 面生成の評価

4.1 節で使用・取得した画像列と特徴点群を利用し、面生成について検証する。マーチンキューブ法により生成された面群が形成する形状は、3次元特徴点をボクセル空間に変換する時の分割数と、しきい値  $S$  により変化する。そこで、分割数としきい値  $S$  をそれぞれ変更した時の生成結果を示す。以後、3次元特徴点を  $N \times N \times N$  に分割する場合、単に分割数は  $N$  とする。

分割数の変更による形状の変化を図 28 に、しきい値  $S$  による変化を図 29 に示す。また、分割数としきい値の関連性について検証するため、異なる分割数により生成される形状について、しきい値を変更することで同等の形状結果を得られるか検証した (図 30)。

図 28 より、分割数が減少すると形状が本来の位置より外側に膨らみ、肥大化するが面どうしの間が埋まり、形状が密になる傾向があることがわかる。一方、図 29 より、しきい値  $S$  が増加すると面が減り、形状は面どうしの間隔が広がり、疎になる傾向があることがわかる。

分割数としきい値  $S$  について、図 30 より異なる分割数の場合ではしきい値をどのように変化させても同一の結果は得られないが、形状が比較的近い結果は得られることが分かった。さらに、分割数を増やし、しきい値を減らすと密だが凹凸の激しい形状になり、逆に分割数を減らし、しきい値を増やすと疎だが滑らかな面による形状復元結果となることが分かった。

次に面にテクスチャを付加した場合について検証する。しきい値  $S$  を 1.0 に固定し、分割数を変更して生成した結果を図 31 に示す。図 31 より、分割数

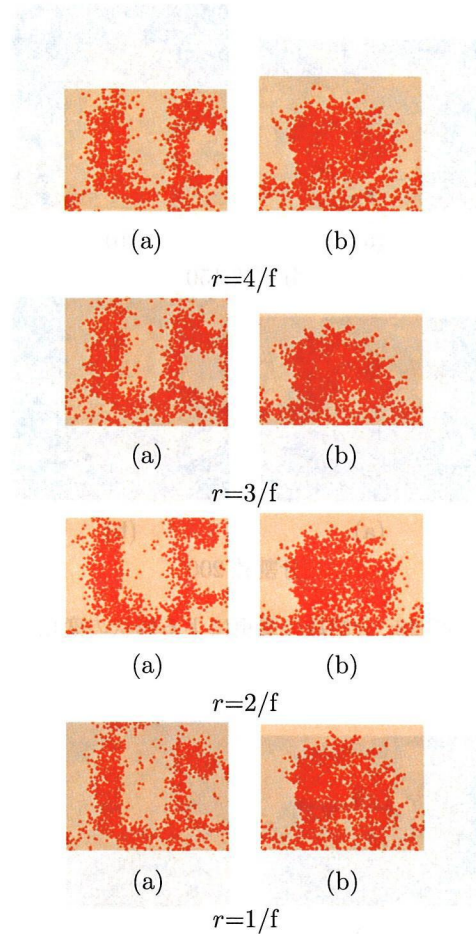


図 26: 半径  $r$  の違いによる形状変化

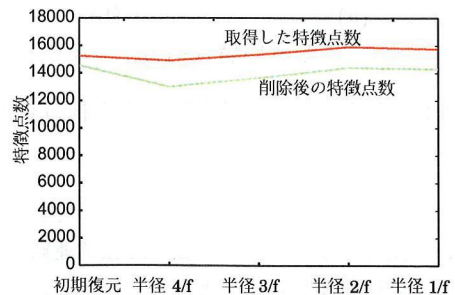
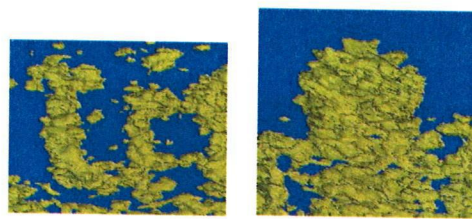


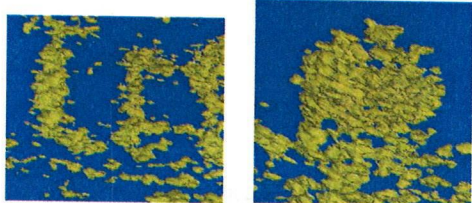
図 27: 半径  $r$  と特徴点数





(a) (b)

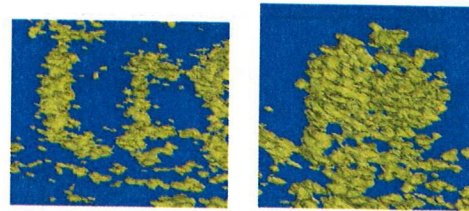
分割数:150



(a) (b)

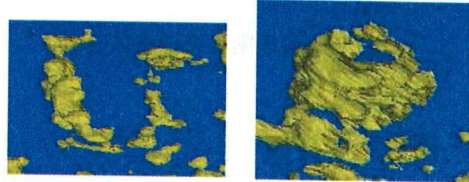
分割数 200

図 28: 分割数の変更による形状の変化



(a) (b)

分割数 200, しきい値  $S=1.0$  の形状

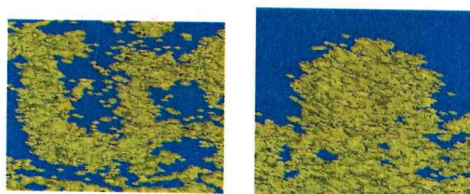


(a) (b)

分割数 100, しきい値  $S=5.0$  の形状

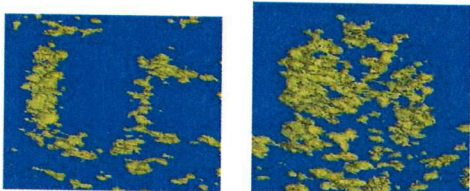
図 30: 分割数としきい値  $S$  の関連性

定すれば良いと推測できる。



(a) (b)

しきい値  $S=0.5$



(a) (b)

しきい値  $S=1.5$

図 29: しきい値  $S$  の変更による形状の変化

が大きいと面が分離し、視覚的にあまり良くない。一方、分割数を小さくすると面が比較的連続するため、視覚的に好ましい結果となる。しかし、ドーム部分が引き延ばされるなど問題も存在する。

このことから、テクスチャを付加する場合には最初は分割数は比較的小さくし、面が連続している範囲内で分割数を大きくしながら、目視で分割数を決

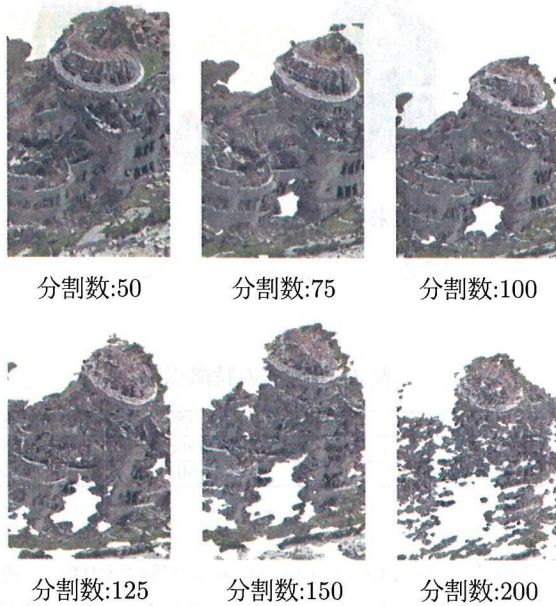


図 31: 分割数の変更におけるテクスチャ付加結果

### 4.3 3次元モデルの生成実験

#### 4.3.1 シミュレーション実験

従来研究の手法による復元結果と本手法による復元結果を示す。実験では図 32 に示すように物体とカメラを配置した。

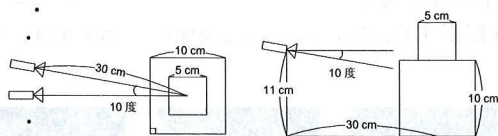


図 32: 物体形状とカメラの位置関係, 上面図(左), 側面図(右)

対象物体は 10cm の立方体の上に 5cm の立方体が置いてあり, 上下の立方体のテクスチャは異なっている。図 32(左) は, この設定を物体上側から見た図であり, 図 32(右) は物体の側面から見た図である。また, 画像は対象物を 10 度刻みで周回しながら 36 枚 (720 × 480 画素) 撮影する。得られたシミュレーション画像の一部を図に示す。

まず, 1 画像につき Harris の検出法により 300 点の特徴点を取得し, 1939 点からなる仮の形状復元

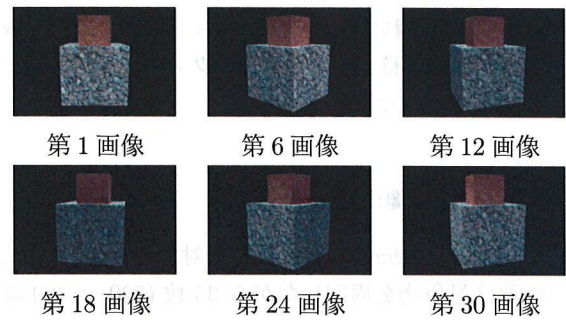


図 33: 使用した画像例:シミュレーション

を行い, ロバスト推定により 100 点の除去を 4 回行うことで, 1539 点からなる形状復元を行った。

この形状復元結果を使用し, 両手法による形状復元をおこなった 3 次元特徴点群と 3 次元モデルを図 34,35 に示す。(a) が従来手法, (b) が本手法である。また, 表 3 に特徴点数を示す。

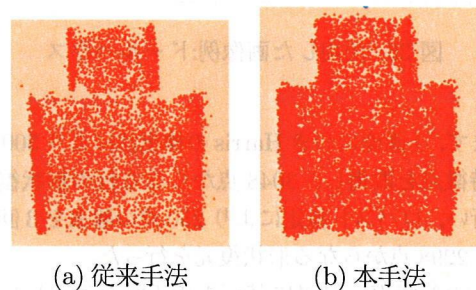


図 34: シミュレーション実験・3次元特徴点群

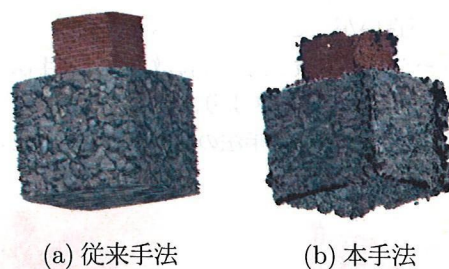


図 35: シミュレーション実験・3次元モデル

図 34 より, 3 次元特徴点群による形状には大きな差はないが表 3 より本手法の方が密に取得できている。しかし, 図 35 より, テクスチャを付加した



3次元モデルでは従来手法の方が面がきれいに示されている。これは、マーチンキューブ法により生成された面群が整っておらず、テクスチャも統一性に欠けるためだと予想される。

#### 4.3.2 実画像:ドールハウス

10 × 8 × 12cm の家の模型を対象とした。なお、画像は対象物を周回しながら 35 枚 (720 × 480 画素) 撮影する。図 36 に得られた画像の一部を示す。

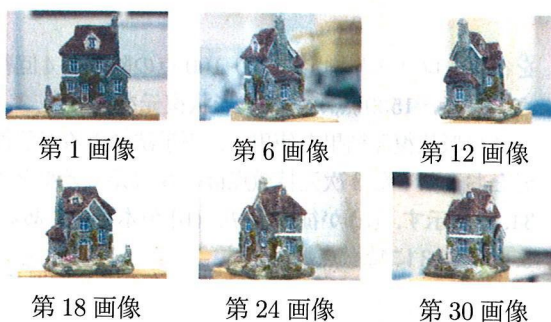
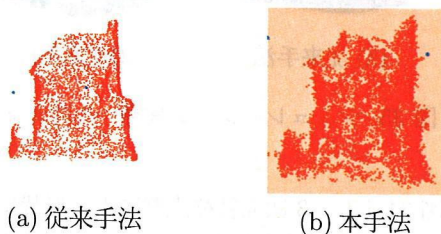


図 36: 使用した画像例:ドールハウス

まず、1 画像につき Harris の検出法により 300 点の特徴点を取得し、3048 点からなる仮の形状復元を行い、ロバスト推定により 250 点の除去を 3 回行い、2298 点からなる形状復元を行った。

この形状復元結果に基づき両手法を適用して生成した形状復元結果と 3 次元モデルを図 37,38 に示す。(a) が従来手法、(b) が本手法による結果である。また表 3 に特徴点数を示す。

図 37 より、密な形状復元における形状は従来手法の方が良いが大きな誤差はない。一方、表 3 シミュレーション実験と同じく特徴点数は本手法の方が多い。しかし、図 38 より、3 次元モデルの生成結果では明らかに従来手法の方が優れた結果となっ



(a) 従来手法 (b) 本手法

図 37: 実画像実験・密な形状復元



(a) 従来手法 (b) 本手法

図 38: 実画像実験・3 次元モデル

表 3: 両手法の特徴点数

	従来手法の特徴点数	本手法の特徴点数
シミュレーション	8965	27503
実画像 ハウス	5665	14493

ている。これはシミュレーション実験と同様に、面の向きが整っていないため、テクスチャの統一性がないためと考えられる。

一方、図 38 の左下の部分は、本手法では建物と分けて表示できているが、従来手法では建物とつながってしまっている。これより、本手法では間に空間を持つ形状を分けて表示できることがわかる。

#### 4.3.3 実画像:昆虫(ヘラクレスオオカブト)

15 × 8 × 3cm の昆虫の標本を対象とした。なお、画像は対象物を周回しながら 35 枚 (720 × 480 画素) 撮影する。図に得られた画像の一部を示す。

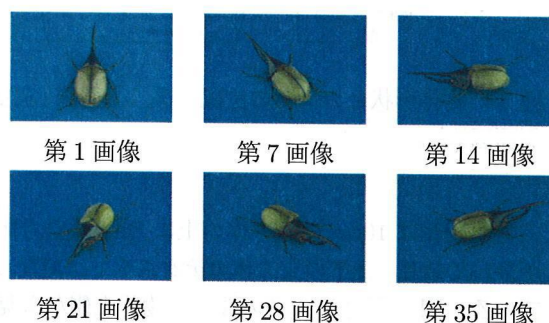


図 39: 使用した画像例:昆虫(ヘラクレスオオカブト)

まず、1 画像につき Harris の検出法により 300 点の特徴点を取得し、2397 点からなる仮の形状復元



を行い、ロバスト推定により 50 点の除去を 7 回繰り返し、2047 点からなる形状復元を行った。

この形状復元により取得した 3 次元特徴点群を基に本手法を 2 回適用し、形状復元を行った。なお、2 回目に基とした特徴点群は、1 回目で取得した特徴点群から 4 枚以上の画像で対応付けされている特徴点を使用した。取得した形状復元結果を図 40 に、3 次元モデルを図 41 に示す。また、特徴点は 1 回目適用時には 18664 点、2 回目適用時には 18977 点取得した。



図 40: 特徴点群による形状復元



図 41: 3 次元モデル

特徴点数は 2 度繰り返してもほとんど増加しなかった。しかし、3 次元形状は特につのと頭部の境目の部分について、1 回目では変形していた形状が 2 回目では本来の形状となった。昆虫の足の部分について、テクスチャはほぼ背景を取得してしまったため足だと分かりにくいですが、6 本の足すべて分かれて表現できている。昆虫の頭部は特徴点がなく、3 次元モデルでも大きな穴が空いている。これは画像上で頭部に特徴点が取れなかったためである。

#### 4.3.4 原爆ドームの復元結果

4.1 節で使用した画像列より、3 次元モデルの生成を行った。

まず、1 画像につき Harris の手法により 300 点の特徴点を取得し、3538 点からなる仮の形状復元を行い、ロバスト推定により 250 点の除去を 9 回繰り返し、1175 からなる形状復元を行った。

この形状復元結果に基づき従来手法と提案手法をそれぞれ適用して生成した形状復元結果を図 42 に示す。(a) が従来手法、(b) が本手法による結果である。

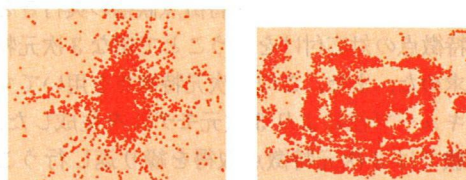


図 42: 原爆ドーム:形状復元結果

図 42 より、従来手法では形状復元が行えなかった。しかし、本手法では形状復元を行うことができた。従来手法で形状復元が行えなかったのは、図 43 に示すような誤対応点が多数取得され、カメラ位置・姿勢の推定が正確に行えなかったためである。

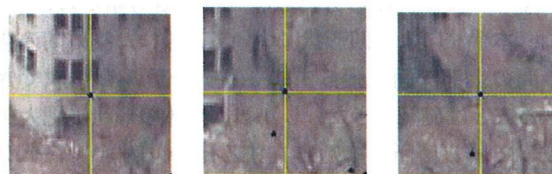


図 43: 誤対応点

本手法で形状復元した結果より生成した 3 次元モデルを図 44 に示す。



図 44: 原爆ドーム:3 次元モデル

## 5 まとめ

本研究では地面などの水平面や凹形状の対象について3次元モデルを生成する手法を提案した。その手法では、すでに求めた3次元特徴点を球と考えることで近傍により新たな特徴点候補の奥行きを求め、特徴点の対応付けを行うことで密な3次元特徴点を求めた。また、この3次元特徴点を用いてマーチンキューブ法により3次元モデルを生成した。

実験では3次元特徴点取得を繰り返し行うことでより密な3次元特徴点を取得できるか検討した。その結果、2回目まではより良い結果が導かれることを示した。しかしそれ以降は条件を変更したが特徴点数が増加しないことや形状が不安定になるなどの問題が発生した。

マーチンキューブ法による面生成では、分割数と面生成の基準となるしきい値  $S$  を変更した時の面の生成結果の特徴を調査した。さらに生成した面にテクスチャを付加することで3次元モデルを生成し、テクスチャを張り付けた時のより良いマーチンキューブにおける分割数についても考察した。

本手法により凹形状について、離れた物体間の空間の上部に面を張らずに3次元モデルを生成できることは確認できた。しかし、全体の形状は肥大化している箇所が多数存在するほか、付加したテクスチャも統一性に欠けるため、これらの問題を解決する必要がある。最初の目的であった地面などの水平面の生成については、原理的には可能となった。よって、特徴をもつ地面は復元可能である。しかし、特徴の少ない地面にはHarrisの検出法では特徴点が取得しにくく、仮に特徴点を取得できても対応付けにおいて誤対応が頻繁に発生するため復元できなかった。このように、特徴点を取得しにくい部分をもつ対象の3次元モデルを生成する手法の開発が今後の課題である。

## 参考文献

- [1] T. Oishi and T. Masuda and K. Ikeuchi: "Digital Restoration of the Cultural Heritages," *Proc. VSMM* pp. 934–941, 2002.
- [2] P. Beardsley, P. Torr and A. Zisserman: "3D Model Acquisition from Extended Image Sequences," *Proc. 4th ECCV'96*, Cambridge, UK., vol. 2, pp. 683–695, 1996.
- [3] Reinhard Koch, Marc Pollefeys and Luc Van Gool: "Multi Viewpoint Stereo from Uncalibrated Video Sequences," *Proc. ECCV'98*, LNCS, Springer-Verlag, Freiburg, pp. 55–71, 1998.
- [4] 佐藤智和, 神原誠之, 横矢直和, 竹村治雄: "マルチベースライン法を利用した動画像からの野外環境の三次元モデル化", 信学技報, PRMU2001-233, pp. 1–8, 2002.
- [5] Z. Zhang, R. Deriche, O. Faugeras and Q.-T. Luong: "A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry," *Proc. 3rd Artif. Intell.*, **78**, pp. 87–119, 1995
- [6] 金澤靖, 金谷健一: "段階的マッチングによる画像モザイク生成", 信学論 D-II, J86-D-II, pp. 816–824, 2003.
- [7] M. Okutomi and T. Kanade: "A multiple-baseline stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 4, pp. 353–363, 1993.
- [8] L. Matthies, T. Kanade and R. Szeliski: "Kalman filter-based algorithms for estimating depth from image sequences," *International Journal of Computer Vision*, vol. 3, pp. 209–239, 1989.
- [9] 保田和隆, 浅田尚紀: "大局的幾何制約を用いた画像列からの高精細3次元モデル生成", 2004年度広島市立大学大学院 情報科学研究科知能情報システム工学専攻 修士論文, 2004.
- [10] C. Harris and M. Stephens: "A Combined Corner and Edge Detector," *Proc. Alvey Vision Conf.*, pp. 147–151, 1988.
- [11] 右田剛史: "視覚情報と接触情報の統合による能動的3次元認識", 広島市立大学大学院情報科学研究科修士論文, 2000.

- [12] William E. Lorensen and Harvey E. Cline:  
“Marching cubes: A high resolution 3D  
surface construction algorithm” *SIGGRAPH*  
*'87Proceedings*, pp163-170, 1987
- [13] 松宮雅俊, 竹村治雄, 横矢直和: “自由形状モ  
デリングのための陰関数曲面を用いた仮想粘土  
細工システム”, 情報処理学会論文誌, Vol.42,  
No.5, pp.1151-1160, 2001

# 線分要素の自動対応付けに基づいた画像列からの疎な3次元形状復元

## 1 はじめに

実物体の3次元モデル化技術は、仮想空間におけるリアルな物体表現として利用されるだけでなく、文化財や芸術作品のデジタルアーカイブなど多様な目的に用いられている重要な技術である。最近では、レーザレンジセンサを用いて高精細な3次元モデルを生成することが行われているが、専用の計測装置が必要であり、対象物の大きさや表面特性などいくつかの条件を満たさなくてはならないという問題がある。このため従来から、迅速かつ手軽に3次元形状を生成できるシステムが必要とされ、コンピュータビジョン技術の応用が研究されている。その1つとして、多視点画像からカメラと対象物の相対運動および対象物の3次元形状を復元する画像列からの3次元形状復元 (Structure from Motion) を用いた3次元モデル生成に関する研究が多く行われている [1]。

画像列からの形状復元を用いた3次元モデル生成では、まず、視点の異なる画像間で対応する点を追跡し、カメラの位置・姿勢と3次元形状の復元を行った後、最終的な3次元モデル生成が行われる。この特徴点の対応付け・追跡を正確に行うことができた場合、透視投影モデルに基づき各点の3次元位置の復元が可能となる。そのため、様々な特徴点の対応付け・追跡や3次元モデル生成に関する手法が提案されている。また我々は、これら個々の技術がもつ前提条件の誤差を許容しつつ、数十枚の画像列から3次元モデルを自動生成する一連の処理システムを提案してきた [1]。しかし、文献 [1] の手法では、生成モデルの線分要素に歪みを生じやすいという問題がある。そこで、文献 [2] では、直線状の輪郭をもった人工物を対象とし、対象物の線分要素を特徴線として扱うことで、歪みの少ないモデルが生成できる手法を提案している。しかし、画像間での特徴線の自動対応付けは困難である。

本研究では、文献 [2] と同様に直線状の輪郭をもった人工物を対象として、その全周囲を撮影した画像列から、物体の角や頂点で表された疎な3次元形状の復元を目的とする。本研究で取り上げる対象物の3次元モデルを生成する場合、密で精細な形状復元を行う必要はなく、物体を構成する代表的な角や頂点による疎な形状復元で十分である。疎な形状復元の場合、一般にTomasi-Kanadeらによる因子分解法 [3] が用いられる。しかし、因子分解法では全特徴点が全画像間で観測されなければならないという制約がある。本研究で

は、全周囲画像列を想定しているため、隠蔽やフレームアウトの影響を受けやすく、因子分解法は適用が困難である。そのため、再投影誤差最小化原理に基づく形状復元を行う。また、特徴点の追跡結果が途切れた場合、同一位置に存在するはずの特徴点と、別々の特徴点として存在してしまう可能性があり、復元結果に影響を与える。本研究では、段階的に特徴点の再追跡を行うことで、追跡途切れによる問題を解決する。

本研究では、まず、物体の角や頂点を表す特徴点として、エッジを含む2値画像から抽出した各線分の交点を用いる。次に、アフィン制約により2画像間を大局的に合わせ、特徴点の対応付けの範囲を限定することで初期の特徴点追跡を行う。この追跡結果と画像をもとに再追跡を行うことで、小数の画像間隔において追跡が途切れていた特徴点同士を連結する。得られた再追跡結果をもとに、非線形最適化計算により、特徴点位置とカメラ位置・姿勢を推定する。推定された特徴点のうち、3次元位置が近接した点同士を統合し、再度推定を行うことで最終的な復元結果とする。

## 2 線分要素を利用した形状復元

画像から抽出された線分要素を利用した形状復元の処理について下記に述べる。まず、物体の全周囲を30～40枚程度で撮影した画像列が与えられる。撮影したカメラの焦点距離は既知であるとする。まず、物体の特徴を表す角や頂点を検出するため、画像から物体の線分要素を抽出する。その処理として、 $\gamma-\omega$  ハフ変換と整列検証法を用いる。次に、抽出された各線分同士の交点を物体の特徴点として求める。次に、アフィン制約により、画像を大局的に位置合わせし、特徴点の初期の対応付け・追跡を行う。また、画像間における特徴点の追跡途切れを考慮し、小数の画像間隔がある場合にも特徴点の追跡が可能となるよう再追跡を行う。この特徴点の再追跡結果を用いて、再投影誤差最小化により特徴点の3次元位置とカメラ位置・姿勢を推定し、仮の形状復元とする。この際、復元精度を向上させるため、得られた復元結果から3次元位置が近接した特徴点同士を統合する。最後に、統合された特徴点に対し、再度形状復元を行うことで、物体の角や頂点が安定した復元結果を得る。

## 2.1 ハフ変換を用いた線分抽出

本研究では、画像から線分を抽出するための手法としてハフ変換を用いる。ハフ変換とは、2値画像などに対して直線や円を抽出する手法であり、基本的な  $a-b$  ハフ変換や、一般的によく用いられる  $\rho-\theta$  ハフ変換がある。 $a-b$  ハフ変換は精度を求める場合、パラメータ空間の大きさが十分でない。また、 $\rho-\theta$  ハフ変換はデジタル画像を対象とした場合、パラメータ空間の歪みによりセルに蓄積される投票数に偏りを生じてしまう。そこで本研究では、パラメータ空間の歪みを補正する方法として、 $\gamma-\omega$  パラメータを利用したハフ変換 [6] を用いる。この  $\gamma-\omega$  ハフ変換は、投票数に偏りがなく、投票軌跡が区分的な直線であるという特徴をもつ。以下にその処理手順を示す。

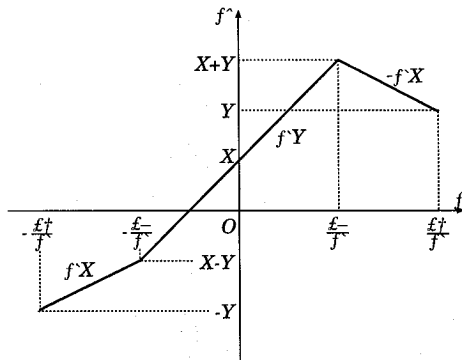


図 1:  $\gamma-\omega$  パラメータ空間における投票軌跡

1. 画像の点の座標  $(x, y)$  を以下の式 (1), (2) を用いて  $\gamma-\omega$  パラメータ空間に投票する。ここで、原点は画像の中心に位置する。また、 $\alpha$  は投票軌跡の傾きとデジタル直線の角度分解能を決める係数であり、本手法では  $1/(\text{画像の横サイズ})$  とする。

$$x = -\alpha\omega y + \gamma \quad \left(-\frac{1}{\alpha} \leq \omega < \frac{1}{\alpha}\right) \quad (1)$$

$$y = (\alpha\omega - 2)x + \gamma \quad \left(\frac{1}{\alpha} \leq \omega < \frac{3}{\alpha}\right) \quad (2)$$

投票軌跡の例を図 1 に示す。ここで、 $X$  は (画像の横サイズ/2)、 $Y$  は (画像の縦サイズ/2) とする。

2. 非極大点の抑制を適用し、極大点を抽出する。
3. 非極大点の抑制で得た極大点を投票数の多い順にソートする。
4. 極大点の投票数の多い順にそれぞれの極大点に對する直線の方程式を得る。

以上の処理により、直線の方程式が得られる。一般に、ハフ変換による直線検出はピーク検出に基づいているため、直線の端点を直接求めるといった線の特性に基づいた検出ができない。本研究では、画像空間を再走査することなく、線分の検証を行うことができる整列検証法 [6] を用いて直線の線分化を行う。

## 2.2 アフィン制約を用いた特徴点の対応付け

アフィン変換行列

$$A = \begin{bmatrix} a & b & t_u \\ c & d & t_v \end{bmatrix}$$

を用いると、画像  $I$  上の座標  $(u, v)$  と画像  $I'$  上の座標  $(u', v')$  の対応は、次式により表される。

$$[u' \ v' \ 1]^T = A[u \ v \ 1]^T \quad (3)$$

2 画像間の最適アフィン変換行列  $A$  は、対応する画素値の差の 2 乗和が最小となるように次式によって求める。

$$\arg \min_A \sum_{(u,v)} |I(u, v) - I'(u', v')|^2 \quad (4)$$

ここで、 $I(u, v), I'(u', v')$  はそれぞれ、画像  $I, I'$  の座標  $(u, v), (u', v')$  での画素値とする。上記の計算は、まず並進成分  $[t_x, t_y]^T$  の最適値を 1 画素毎の全探索により求め、次にその値を初期値として、6 個全てのパラメータを勾配法により求める 2 段階の探索により行った。なお、局所解を避けるため画像を平滑化し低周波成分によるマッチングを行う。実際の処理手順は、下記の通りである (図 2)。

1. 画像  $I, I'$  の両方において、2.1 節で述べた方法により抽出した直線を線分化し、各線分の画像上の交点を求める。この求めた交点を物体の特徴点として利用する。
2. アフィン変換前の画像において、画像  $I$  の 1 つの特徴点  $P$  に対して、画像  $I'$  の特徴点の中で、周辺画素同士の画素値の残差 2 乗和 (SSD) を評価基準として、最も類似しているものから順に  $m$  個の点  $Q_1 \sim Q_m$  を選択しておく。
3. 2 画像間で求めたアフィン変換行列を用いて、画像  $I$  の特徴点  $P$  を画像  $I'$  上に変換する。変換した点を  $P'$  とする。
4.  $Q_1 \sim Q_m$  のうち、 $P'$  の近傍領域に含まれるものがある場合、それを対応点とする。複数ある場



合, その中で SSD の評価基準により最も類似している点と対応付ける. 近傍領域に含まれる点がない場合, 点  $P$  の対応点はないものとし, 特徴点から除外する.

以上の 1. ~ 4. までの処理を, 画像  $I$  の全ての特徴点に対して順に行う. この際, 画像  $I$  の特徴点  $P$  は, 対応する特徴点の第 1 候補  $Q_1$  の SSD による評価値が最も良い (類似している) ものから順に選択する. また, 特徴点  $P$  と対応付けられた特徴点  $Q_{i^*}$  ( $i^* \in 1 \dots m$ ) は, 以降の対応付けからは除去する. この手法は, 従来から行われている貪欲法に, アフィン制約を追加したものとなっている. これにより, 大局的な幾何制約を考慮した特徴点追跡を行う.

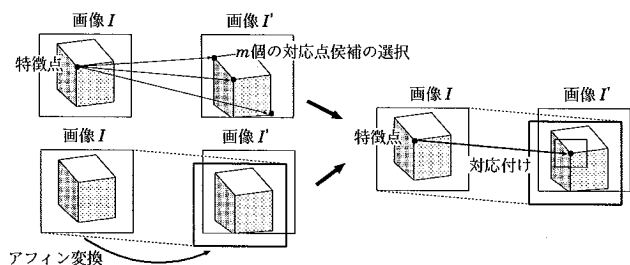


図 2: アフィン制約を用いた特徴点の対応付け

### 2.3 特徴点の再追跡

2.2 節の処理によって得られた特徴点の追跡結果から, 画像間で途切れた点を連結することで特徴点の再追跡を行う.

まず, 各特徴点の画像間における追跡の終了画像  $I$  とその点  $P$  に注目する (図 3). 次に, 画像  $I$  から  $n$  枚先の画像において, 他の特徴点  $P'$  による追跡の先頭画像  $I'$  の存在を判定する. 先頭画像が存在する場合, 画像  $I$  の点  $P$  と  $I'$  の点  $P'$  に対し, 2.2 節と同様の処理を行い, 対応点を探索する. 対応点が見つかった場合, 点  $P$  として統合する. 複数の画像において対応点候補が存在する場合, 画像間における追跡数が最大のもので対応点として採用する. ここで, 対応点の追跡の終了画像が点  $P$  を含む画像を越えた場合, 対応点から除去する. また, 先頭画像が存在しない場合, 点  $P$  の追跡対応点はないものとし, 画像間における点  $P$  の追跡が 3 枚より少ない場合, 信頼性の低い特徴点として除外する. 以上の処理を, 全特徴点の追跡結果について繰り返す.

これにより, 特徴点の追跡が隣接画像間だけに限定されず, 小数の画像間隔がある場合においても追跡することが可能となる. また, 同一位置の特徴点を別々

の特徴点として推定されることが減るため, 安定した形状復元ができる.

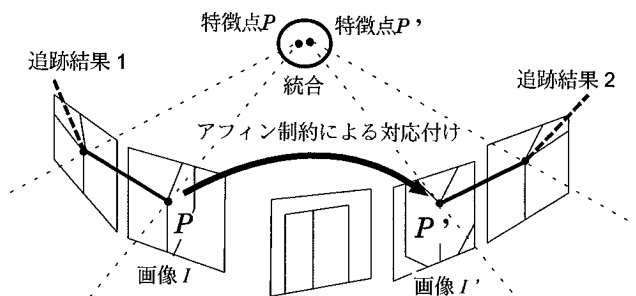


図 3: 小数の画像間隔を考慮した特徴点の再追跡

### 2.4 再投影誤差最小化による形状復元

以上の処理によって得られた画像列全体の特徴点対応を用いて, 物体の形状復元と各画像を撮影したカメラの位置・姿勢パラメータ推定を行う. これは, 「運動からの形状復元」問題として多くの研究がなされているが, 本研究では, 文献 [4] の再投影誤差最小化原理に基づく方法を用いて物体形状とカメラ位置・姿勢の同時推定を行う.

まず, 2.3 節から得られた特徴点の再追跡結果に対して, 仮の形状復元を行う. 次に, 仮の復元結果における各特徴点の 3 次元位置を評価する. 既に 2.3 節の処理により, 信頼性の低い特徴点がある程度除去されていると仮定した場合, 各特徴点のうち 3 次元位置が近接した特徴点同士を統合することで, 復元された 3 次元空間中において同一点の複数存在がより少なくなる. 統合した特徴点に対し, 再度形状復元を行うことにより, 特徴点数は減るが, 物体の角や頂点が安定した復元結果を得ることができる.

## 3 シミュレーション実験

### 3.1 実験環境

図 4 に示すように物体とカメラを配置した.

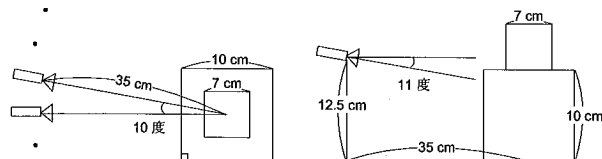


図 4: 物体形状とカメラの位置関係

仮想的な空間に、対象物体は1辺10 cmの立方体の上に1辺7 cmの立方体が置いてあり、上下の立方体のテクスチャは異なっている。カメラ焦点位置と物体中心位置までの距離は35 cmとする。図4(左)は、この設定を物体上側から見た図であり、図4(右)は、物体の側面から見た図である。カメラは、物体を中心として円軌道上を移動しながら、一定間隔で撮影を行う。カメラ位置は下面から12.5 cm上側にあり、物体中心に向けて11度下向である。なお、物体の各面は全て直交している。

以上の環境において、全周囲を10度刻みで36枚撮影する。なお、画像サイズは720×480画素、画像フォーマットはPPM形式である。得られたシミュレーション画像の一部を図5に示す。これらの画像列を用いて次の実験を行った。

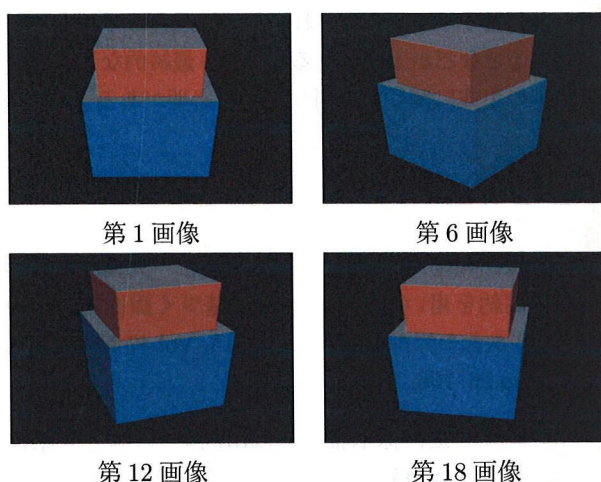


図5: 実験画像例

### 3.2 実験結果

アフィン制約を用いて特徴点の対応付け・追跡を行い、形状復元した結果を図6に示す。この実験では、エッジ検出された2値画像から抽出した線分同士の交点を物体の特徴点として扱い、アフィン変換により対応を求めた画素から30画素以内の範囲で特徴点の対応付けを行った。なお、全ての画像から抽出された線分の総数は516本、それらの交点のうち特徴点として扱われる交点の総数は436点となった。中央の点が3次元空間中の特徴点を表し、周囲の円弧上の点がカメラ位置、カメラ位置から中央の特徴点群に向かう線がカメラ光軸をそれぞれ表している。

図6より、アフィン制約による特徴点の追跡結果として48点が得られた。この追跡結果を利用した形状復元を行った結果、複数の特徴点が同一位置付近に多く推定されているが、全体の大まかな形状は復元され

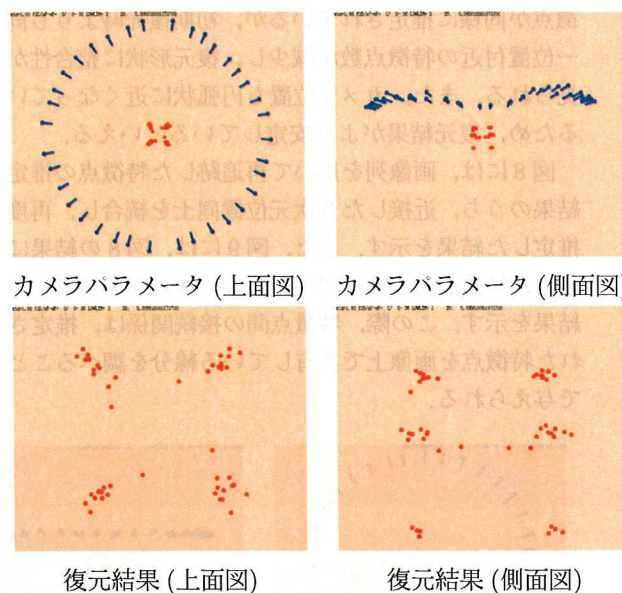


図6: アフィン制約を用いた初期復元結果(48点)

ている。また、カメラ位置はおおよそ円弧状になっており、安定に復元できている。

次に、アフィン制約を用いた初期復元後、画像間において特徴点を再追跡し、形状復元を行った結果を図7に示す。この際、追跡の対応点を探索する画像は2画像先までとした。2画像より多い場合、すなわちカメラ視差が20度より大きくなる場合、対応付け処理が困難となってしまうためである。

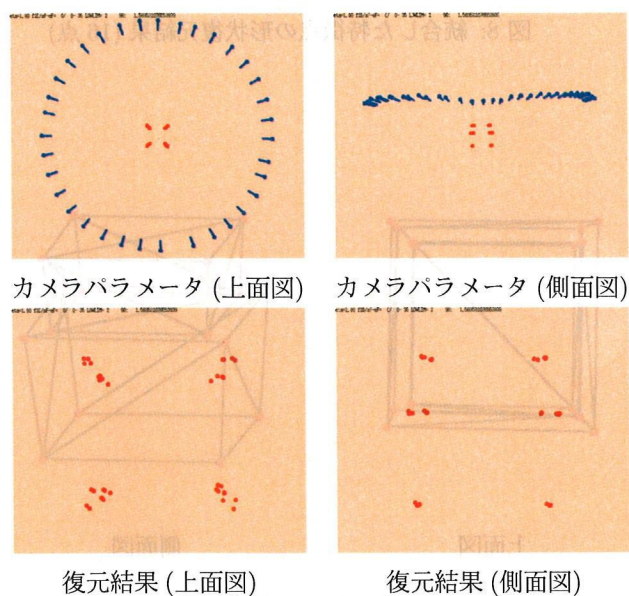


図7: 特徴点の再追跡結果を用いた形状復元結果(30点)

図7では、画像列を用いた特徴点の再追跡結果として30点が得られた。図6と比較した場合、複数の特



特徴点と同様に推定されているが、初期復元時よりも同一位置付近の特徴点数が減少し、復元形状に整合性が見られる。また、カメラ位置も円弧状に近くなっているため、復元結果がより安定しているといえる。

図8には、画像列を用いて再追跡した特徴点の推定結果のうち、近接した3次元位置同士を統合し、再度推定した結果を示す。また、図9には、図8の結果に対し、特徴点間の接続関係を用いて3角面を構成した結果を示す。この際、特徴点間の接続関係は、推定された特徴点を画像上で共有している線分を調べることによって与えられる。

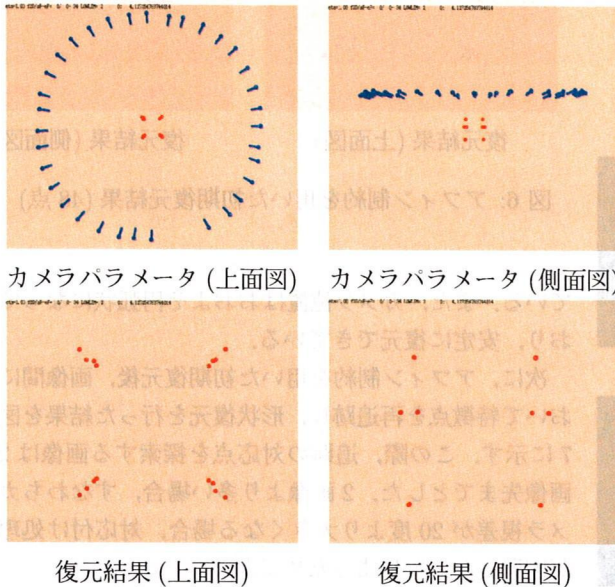


図8: 統合した特徴点の形状復元結果 (16点)

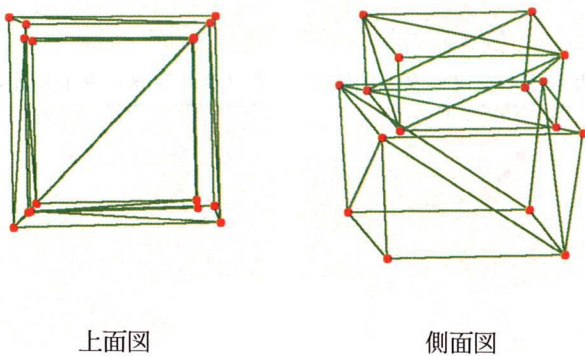


図9: 3角面構成結果

図8と図7の結果を比較した場合、カメラ位置・姿勢に大きな変化は見られないが、推定された特徴点数は16点と対象物の頂点数と等しく、各特徴点の位置

付近において複数の特徴点が存在していなことがわかる。また、図9の結果からも、整合性のとれた復元結果が得られていることがわかる。以上の結果から、本手法により画像間における特徴点の追跡結果の途切れを段階的に連結できていることが確認できた。

## 4 おわりに

本研究では、直線状の輪郭をもった人工物を対象として、その物体の角や頂点で表された疎な形状復元を行う手法を提案した。実験結果より、段階的に特徴点を再追跡し連結することで、安定した形状復元結果を得たことを確認できた。今後の課題として、今回はシミュレーション画像列を用いた実験のみであったが、実画像列を対象にして実験を行うことで本手法の有効性を評価することや、それに伴い、現在の手法を改善することなどが上げられる。また、最終的な3次元モデルの生成に関して検討していく必要がある。

## 参考文献

- [1] 保田和隆, 椋木雅之, 浅田尚紀: “大局的幾何学制約を用いた特徴点追跡に基づく画像列からの3次元モデル生成”, CVIM15, Vol.47 No.SIG10, pp.96-106, 2006. 7.
- [2] 藤村一之, 椋木雅之, 浅田尚紀: “線分要素に基づく画像列からの3次元形状モデルの生成”, FIT2006, 情報科学技術レターズ, vol.5, LI-012, pp205-206, 2006.
- [3] C. Tomasi, T. Kanade: “Shape and motion from Image Stereams under Orthography – A Factorization Method,”, *Int .J. Comput. Vision*, 9-2, pp. 137-154, 1992.
- [4] 右田剛史, 天野 晃, 浅田尚紀: “3次元形状・運動復元のための高速非線形最適化計算法”, 情処論, vol.44, No.11, pp.2864-2872, 2003.
- [5] 椋木雅之, 右田剛史, 青山正人, 浅田尚紀: “非線形最適化による建物画像列からの全周形状一括復元のための初期値設定法”, 情処 CVIM 論, vol.45, no.SIG13(CVIM10), pp.64-73, 2004.
- [6] 和田俊和, 藤井高広, 松山隆司: “ $\gamma - \omega$  ハフ変換-可変標本化による  $\rho - \theta$  パラメータ空間のひずみの除去と投票軌跡の直線化”, 信学論 (D-II), Vol.J75-D-II, No.1, pp.21-30, 1992.