移動ロボット搭載のカメラを用いた物体の三次元形状モデリング

山崎 公俊^{*1} 友納 正裕^{*2} 坪内 孝司^{*1} 油田 信一^{*1}

3-D Object Modeling by a Camera Mounted on a Mobile Robot

Kimitoshi Yamazaki^{*1}, Masahiro Tomono^{*2}, Takashi Tsubouchi^{*1} and Shin'ichi Yuta^{*1}

This paper describes 3-D shape modeling performed by a mobile robot. We propose a method in which the robot in the real environment acquires the accurate shape and constructs a 3-D model of an object using Computer Vision techniques under the condition that position and shape of the object is unknown. A 3-D model is reconstructed from image streams which are captured by a camera mounted on the robot. The experimental results show the effectiveness of our method.

Key Words: 3-D Object Modeling, Shape Reconstruction, Robot Vision, Mobile Robot

1. はじめに

本論文は、実環境を移動するロボットが、環境中の物体の三次元形状モデルを自律的に生成する手法に関して述べるものである。単眼カメラを搭載した移動ロボットを用いて、物体の周囲を走行しつつ取得した多数枚の画像から物体の精密な形状モデルを獲得する。

ロボットが環境中の物体の形状モデルを持つことで、その物体に対する直接的な操作や認識が可能になる。その意味で、環境中の物体のモデルを持つことは、ロボットの行動能力の向上に重要な要素である。従来の研究では、あらかじめ定義した物体の形状モデルをロボットに事前に与えておき、実環境を移動するロボットがモデルを用いて物体を操作する例 [6] や、環境中のナビゲーションに利用する例 [11] などがある。

しかし、我々が普段生活している実環境には非常に多くの物体が存在しており、またその環境は時々刻々と変化する。ロボットに物体操作などの日常の作業を想定した場合、必要なすべての物体の形状モデルをあらかじめ持たせておくことは非常に手間のかかる作業である。このとき、ロボット自身が物体のモデルを生成できれば、実環境で動作する移動ロボットにとって有用である。

筆者らは、物体を把持・運搬し作業を行う自律移動ロボットの実現に興味がある。本論文では要素技術の一つとして、実環境に存在する物体の三次元形状モデルを自律的に生成する機能

を開発する。このとき、より多くの物体にモデリングの機能を適用できる処理システムの構築を目指す。対象とする物体の形状は未知とし、処理の助けとなる人工マーカは与えないことを前提とする。

実験機として、車輪型の移動ロボットを用いる。外界センサとしては、単眼のカメラ（パンチルトカメラ）を移動ロボットに搭載して用いる。形状モデリングでは、ロボットが対象物の周囲を走行しながら取得した画像列を用いて、因子分解法、モーションステレオ、Space Carving などのコンピュータビジョンの手法を利用し、対象物の密な形状を生成する。提案手法では、最初に、オドメトリなどのロボットの動きデータから、物体とカメラとの相対位置を確定する。これを利用し、以後の物体形状とカメラの姿勢変化を、画像列のみから推定する。

以下では、2章で関連研究について述べ、3章で本研究の方針と問題設定について述べる。4章では三次元形状モデリングの概要、5章では提案する密な形状復元手法の詳細、6章で提案手法を実装した実験結果を示す。7章をまとめとする。

2. 関連研究

2.1 三次元形状復元

本研究では、ロボットに単眼のカメラを搭載し、得られる画像を用いて形状モデリングを行う。複数の画像データから物体の三次元形状を復元する研究は、コンピュータビジョンの分野で盛んに行われている。

因子分解法 [1] [12] やエピポーラ幾何を用いた手法 [18] は、複数の画像データから物体の疎な三次元形状とカメラの位置・姿勢を得ることができ、画像間でテクスチャの対応が付く場合には有用な手法である。また、物体の密な形状を復元する手法として、相関ステレオ法 [9] や Space Carving [7] などがある。こ

原稿受付 2004 年 10 月 15 日

^{*1}筑波大学システム情報工学研究科

^{*2}筑波大学

^{*1}System & Information Engineering, University of Tsukuba

^{*2}University of Tsukuba

れらは、カメラの姿勢が既知である場合に利用できる。

一方、種々の手法を統合して、画像列からカメラの動きデータと物体の密な形状を復元する研究も行われている。Sato ら [14] は、あらかじめ指定したマーカと自動で追跡した特徴点からカメラ姿勢を求め、それらを元にマルチベースラインステレオ法、Voxel 空間への統合を行い、屋外環境の密な三次元形状モデルを生成した。Sainz ら [13] は、透視投影モデルに基づく因子分解法により求めたカメラの姿勢を用いて、単純な背景のもとで Space Carving を行い密な形状を得た。

これらの研究例では、対象物にマーカをつけたり、人工的な背景を用いたりという条件のもとで、三次元形状を精密に復元している。これに対し本研究では、実環境を背景に、物体には人工的なマーカがない状態から移動ロボットが自動で三次元形状モデリングを行う。

2.2 環境モデル構築

本研究は、未知・もしくは断片的な情報しかない状態から、対象物の形状とカメラ姿勢を画像列より求めるものである。この枠組みは、移動ロボットの動作モデルと搭載したセンサから得たデータを基に、ロボットのポジショニングと環境地図の構築を同時に行う SLAM (Simultaneous Localization And Mapping) と共通する面がある。SLAM は、近年盛んな研究分野であり、レーザレンジファインダを用いた例 [15] や、ビジョンセンサを用いた例 [8] など、多くの成果がある。

SLAM では、環境の大域的な情報を得ることに重点が置かれている。移動ロボットの走行環境において、ナビゲーションなどを考慮した情報獲得を主目的としたアプローチである。一方本研究では、物体操作を行うための情報獲得を目的としている。環境中の一つの物体を集中的に観察し、密な三次元モデルを獲得する。このアプローチにより生成するモデルは、ロボットの移動環境を広範囲にモデリングする SLAM と比較し、絶対誤差が少なく、詳細な物体形状を表現できる。

3. 問題設定と処理の方針

3.1 問題設定と課題

本研究では、車輪型移動ロボット (Fig. 1) が床面上を移動することにより、複数の視点から画像列を取得し、それらを用いて形状モデリングを行う。ここで獲得する物体の三次元形状モデルは、以後、その物体の拾い上げなどといった物体操作へ利用することを想定する。

三次元モデルの復元精度には、移動ロボットに搭載したロボットハンドで物体を把持する際、形状誤差を把持動作で吸収できる程度の正確さを求める。本研究では、対象物を 20 [cm] 程度の大きさとし、モデリングの形状誤差 1 [cm] 程度を目標とする。

モデリングの対象とする物体に関して、以下の前提をおく。

- (1) ロボットは対象物の形状に関する事前知識を持たない
- (2) 対象物には人工的なマーカを付けない
- (3) 平面だけでなく曲面を含んだ物体形状も対象とする
- (4) 対象物を撮影した画像から特徴的なテクスチャが得られる
- (5) 対象物はロボット前方の床面上に存在する

(1)～(3) の前提条件は、対象物を多面体に制限したり、マーカをあらかじめ付加するような従来の研究例と比べ、困難な問題

Fig. 1 Mobile robot

設定であると言える。(4) は、対象物の形状を獲得する際、画像特徴点を利用するための条件である。画像中のテクスチャから特徴点を抽出し、別の画像との間でその対応を付け、画像間での特徴点の推移を記録する。

また、実環境を移動するロボットにおいて、搭載したカメラ姿勢を直接的に知る手段は、オドメトリによるロボットの自己位置とパンチルトカメラのエンコードによる角度情報である。しかし、ロボットの動きが複雑な場合、これらから得るカメラ姿勢は、画像取得とオドメトリ取得の同期のずれや車体のスリップ、カメラの角度分解能などの要因により誤差を含む。このとき、オドメトリ情報を直接的に用いて画像列による形状復元を行っても、あまり精度の高い形状が得られない場合が多い。したがって、オドメトリなどから得たカメラ姿勢に補正処理を加えるか、あるいはそれらを用いずに形状をモデリングするかの必要がある。復元処理にはオドメトリを用いず画像のみを使う。なお、本論文でのカメラ姿勢とは、位置と角度を含んだカメラの状態表現を意味する。

これらの条件のもとで、物体の精密な形状モデルを獲得することが本研究の課題である。

3.2 処理の方針

本研究では、全体の処理の流れを 2 段階に分ける。

ロボットは処理開始とともに、床面を探索しながら走行する。このとき、環境中には複数の物体が存在してもよいとし、この場合、モデリングの対象物はロボットが自律的に決定する。そして、対象物とロボットとの位置関係を正確に求める。その後、対象物の詳細な形状を得るために、その物体を集中的に観察し、形状モデリングを行う。

(1) 対象物の発見

ロボットは床面上の物体の有無を探索しながら走行する。問題設定より、対象物はロボット前方に存在するため、ロボットの走行経路として、単純な直進経路を与える。得られた画像列に対し、カメラの視野内に存在する物体から対象物を決定し、ロボットとの正確な相対位置を求める。手法の詳細は文献 [16] に譲り、ここでは流れを簡単に述べる。ロボットは、カメラの角度を一定にした状態で直進走行し、取得した画像列から特徴点の抽出・追跡を随時行う。最初の画像と現在の画像間で十分な視差が得られたところで、

Fig. 2 Robot motion

モーションステレオを用いて復元処理を行う。このとき、パンチルトカメラに角度の変化はなく、ロボットの走行は単純なため、オドメトリ等から得られるカメラ姿勢はほぼ正確であるとみなせる。そこで、これらの値をカメラ姿勢として利用する。復元する物体のスケールは、この処理で決定される。カメラ姿勢の若干の誤差に対しては、非線形最小化を行い形状の復元結果とともに補正する。以上の流れにより、床面の物体の有無とそれぞれの三次元位置が判明する。その後、対象物を特定するために、この三次元復元結果から背景情報を除去する。また、複数物体が存在する場合には、復元点群をその床面位置座標でクラスタリングし、ロボットに一番近い点群をモデリングの対象物として決定する。

(2) 対象物の形状モデリング (Fig. 2)

発見した対象物について、その周囲から取得した画像列を利用し、物体の密な形状モデルを生成する。ロボットは、対象物に接近した後その周囲を走行する。ここでは、ロボットの走行経路やカメラの角度変化が複雑になる。そのため、カメラ姿勢と物体形状をオドメトリ等の直接的な情報を用いずに画像列から求める手法を提案する。本論文では、この形状モデリングの詳細について述べる。

4. 形状モデリングのアプローチ

対象物の発見処理により、対象物との相対的な位置関係が把握できる。形状モデリング処理の目的は、ロボットに搭載したカメラにより得た画像列を利用し、対象物の精密な形状モデルを生成することである。

4.1 ロボットの動作と画像からの情報抽出

形状モデリングにおけるロボットの走行経路を以下のようにする。ロボットは、まず発見した対象物に接近する走行を行う。その後、カメラを対象物の方向へ向けながら、その周囲を走行する (Fig. 2)。

このとき、ロボットに搭載したカメラから画像を取得していく。対象物を周囲から観察した画像列が生成されるため、これらより、画像特徴点を抽出・追跡する。本研究では、KLT-Tracker [5] を用いる (Fig. 3)。このトラッカーは、画像の局所領域から、コーナーやテクスチャの強い部分などを抽出し、その特徴が別の画像でどこに移動したかを追跡することができる。

Fig. 3 Feature point extraction

4.2 処理の流れ

本研究での形状モデリングは、以下の2段階に分けられる。

- (1) カメラ姿勢と物体の疎な形状の復元
- (2) 物体の密な形状の復元

カメラ姿勢と物体の疎な形状の復元では、各画像を取得した際のカメラ姿勢と、KLT-Trackerにより対応付けた特徴点の三次元位置を求める。前述したとおり、ここではオドメトリなどから推定するカメラ姿勢に誤差が多く含まれる。そこで、新たに画像を取得するたびに、カメラ姿勢と物体の形状を画像から求める。ロボットの走行中に実行可能な処理を目指し、高速性も重視する。この詳細は次章に述べる。

密な形状の復元は、上述の処理により得たカメラ姿勢と、KLT-Trackerより得た特徴点の推移情報、画像データの色情報を利用して行う。物体をサーフェスモデルとして復元し、密な形状モデルを高精度に獲得する。この詳細は6章で述べる。

5. カメラ姿勢と物体の疎な形状の復元

画像間で対応付けた特徴点の推移情報に対し、カメラ姿勢と物体の疎な形状を獲得する手法を提案する。

5.1 課題とアプローチ

本研究では、ロボットが実環境で物体形状を復元するため、以下の要件を満たす必要がある。

- ロバスト性
実環境で物体を観察してモデリングするため、特徴点追跡のアウトライアや検出位置誤差に頑強である必要がある。
- 処理速度
ロボットが走行しながらモデリングできるために、リアルタイムあるいは準リアルタイムで処理できる必要がある。
- 復元精度
物体をハンドリングする目的に対し、十分な復元精度を有する。本研究では、精度目標を、10~20 [cm] 程度の大きさの物体に対して誤差 10 [mm] 以内とする。
- 処理の逐次性
形状復元手法には逐次型とバッチ型があるが、本研究では疎な形状復元は逐次的に行う。これは、将来、モデリングのための経路計画や視点選択をモデリングの最中に行うことを可能にするためである。
これらの要件を満たすために、既存のコンピュータビジョンの手法を次のように組み合わせる。

(1) 初期復元

初期状態では、カメラ姿勢も物体形状も分からないので、両者を同時に求めることができる因子分解法を用いる [1]。因子分解法は、カメラモデルを弱透視投影などで線形近似するので、非常に安定して解を求めることができ、ロバスト性を高めるのにも有効である。

ただし、カメラモデルの線形近似により、復元誤差が生じるので、非線形最小化によって補正する。これは、復元形状の画像への逆投影誤差が最小になるようにカメラ姿勢と物体形状を微調整するものであり、bundle adjustment と呼ばれて、誤差修正のために広く用いられている [2]。

(2) 逐次復元

カメラの初期姿勢と物体の初期形状が求まれば、それをもとにして、次の画像に対するカメラ姿勢と物体形状を求めることができる。このために、まずそれまでに復元した三次元点と現在画像の特徴点とを対応させて次のカメラ姿勢を計算する。ここでも逆投影誤差に関する非線形最小化を用いる。

次に、得られたカメラ姿勢を用いて、モーションステレオにより、特徴点 1 点ごとにその三次元位置を復元する。この際、各特徴点ごとに復元誤差を評価し、誤差の大きい点は復元結果に含めないようにする。

逐次復元フェーズを因子分解法で実現することも可能であるが、視点変化に伴う特徴点の入れ替わりの対処がやや煩雑である [3]。また、因子分解法のように、画像特徴点のみから復元する場合は、カメラの並進成分（基線長に相当）が十分大きくないと退化を起こすことがある。一方、上記の非線形最小化でカメラ姿勢を求める方法では、三次元点と二次元点の対応に基づくので、カメラ運動の並進成分が小さくても安定して解が得られる。そのため、提案手法では、初期復元にだけ因子分解法を用いる。

実画像では特徴点追跡で大きなアウトライアが生じることがあるため、その除去に RANSAC などのロバスト推定を行う。その際、現在仮説のカメラ姿勢に対して、特徴点の復元誤差を 1 点ずつ評価してアウトライアかどうかを判定する。この処理には、特徴点を 1 個ずつ扱えるモーションステレオが有用である。また、モーションステレオは計算が簡単であり、非常に高速な復元が可能である。

処理の流れを Fig. 4 に示す。以下、5.2 節、5.3 節で、提案手法の詳細について述べる。

5.2 初期復元フェーズ

本研究で提案する形状モデリングでは、ロボットはまず、モデリングの対象物に近づくように走行する。前述したように、ここではオドメトリなどから得るカメラ姿勢の誤差が大きくなるため、画像情報だけを用いて物体形状とカメラ姿勢を求める。このために、因子分解法を用いる。

まず、対象物に近づきつつ取得した複数枚の画像から、複数の特徴点を対応付け因子分解法を行う。本研究では、安定した結果を得るため、カメラモデルを弱透視投影モデルに近似した因子分解法 [12] を適用する。具体的には、画像特徴点の推移データを並べた計測行列 $\mathbf{W}$ を作成し、特異値分解を用いてカメラ

Fig. 4 Camera pose and sparse shape reconstruction

運動行列と物体形状行列に分解する。

$$\mathbf{W} = \mathbf{M}\mathbf{S} \quad (1)$$

これより、行列 $\mathbf{M}$ としてカメラの運動が、行列 $\mathbf{S}$ として特徴点の三次元位置が得られる。

ロボットが対象物に近づく間は、取得する画像列にオクルージョンが起こりにくく、多くの特徴点が複数枚の画像で観測し続けられる。このため、特徴点の消滅処理を考慮する必要がなく、因子分解法を容易に適用できる。

ただし、因子分解法の結果にはカメラモデルの線形近似誤差が含まれるため、非線形最小化によって誤差を補正する。具体的には、因子分解法で得た結果を初期値として、以下の評価関数を最小とするカメラ姿勢と物体形状を求める。

$$C = \sum_{j=0}^2 \sum_{i=0}^P \left(\frac{\mathbf{r}_{xj}^T \mathbf{m}_{ij}}{\mathbf{r}_{zj}^T \mathbf{m}_{ij}} - \frac{X_i + t_{xj}}{Z_i + t_{zj}} \right)^2 + \sum_{j=0}^2 \sum_{i=0}^P \left(\frac{\mathbf{r}_{yj}^T \mathbf{m}_{ij}}{\mathbf{r}_{zj}^T \mathbf{m}_{ij}} - \frac{Y_i + t_{yj}}{Z_i + t_{zj}} \right)^2 \quad (2)$$

$\mathbf{m}_{ij}$ は、 j 枚目の画像における i 番目の特徴点の投影座標を表す。 $\mathbf{r}$ は回転行列 $\mathbf{R}$ の列ベクトルであり、 t_x, t_y, t_z はワールド座標系からカメラ位置への平行移動成分、 X, Y, Z は特徴点の三次元座標を表す。因子分解法の結果は、非線形最小化の初期値としては十分に良い値であり、この非線形最小化の収束は一般に速い。

以上により、物体の一部の疎な形状とカメラ姿勢の初期値を精度良く求めることができる。

5.3 逐次復元フェーズ

前述した流れで生成した初期値を利用して、物体の全周囲の形状と、画像取得時のカメラ姿勢を復元する。

具体的には、新しく画像を取得するたびに以下の処理を行う。

- 三次元位置が既知で、画像間で追跡が続いている特徴点から、カメラ姿勢を非線形最小化により推定

- 求めたカメラ姿勢から、新たに抽出された特徴点の三次元位置をモーションステレオにより算出

カメラ姿勢は、新たに取得した画像から追跡した特徴点のうち、三次元位置が既知の点を用いて算出する。具体的には、現在のカメラ姿勢を変数とし、一つ前のカメラ姿勢を初期値とした非線形最小化を行う。

$$C = \sum_{i=0}^P \left(\frac{\mathbf{r}_x^T \mathbf{m}_i}{\mathbf{r}_z^T \mathbf{m}_i} - \frac{X_i + t_x}{Z_i + t_z} \right)^2 + \sum_{i=0}^P \left(\frac{\mathbf{r}_y^T \mathbf{m}_i}{\mathbf{r}_z^T \mathbf{m}_i} - \frac{Y_i + t_y}{Z_i + t_z} \right)^2 \quad (3)$$

このフェーズでは、オクルージョンの影響が大きくなるため、画像間での特徴点の消滅や生成に対処する必要がある。このためには、特徴点の三次元位置算出を1点ごとに行えるモーションステレオが有効である。モーションステレオでは、以下の式を最小化することで、特徴点の三次元位置を求める。[4]

$$C = \|\mathbf{X} - s_1 \tilde{\mathbf{m}}_1\|^2 + \|\mathbf{X} - s_2 \mathbf{R} \tilde{\mathbf{m}}_2 + \mathbf{T}\|^2 \quad (4)$$

$\tilde{\mathbf{m}}_1$, $\tilde{\mathbf{m}}_2$ は、2枚の画像における対応点の投影座標の拡張ベクトルである。 $\mathbf{X} = (X, Y, Z)$ は特徴点の三次元位置、 $\mathbf{R}$ は2枚の画像間でのカメラの相対姿勢、 $\mathbf{T}$ は相対位置である。特徴点の復元精度は、求めた三次元位置を現在の画像上に逆投影し、その投影位置と KLT-Tracker による特徴点の追跡位置との距離（逆投影誤差）によって判定する。逆投影誤差が所定の閾値以下であれば、その特徴点を既知点として登録し、以降のカメラ姿勢の推定に利用する。

この処理において、KLT-Tracker が追跡ミスを起こした特徴点をもとにカメラ姿勢の推定を行うと、非線形最小化の処理が収束しないことがある。そこで、本研究では、RANSAC [10] を用いてアウトライアを除去し、ロバストなカメラ姿勢の推定を行う [17]。

6. 密な形状獲得

前章の結果を用い、物体の密な形状復元を行う。本章では、その手法について述べる。

6.1 課題とアプローチ

密な形状を求めるため、Space Carving [7] を行う。本章では、画像間で特徴点の対応が既知であることを利用した Space Carving の高速化手法も提案する。

Space Carving は本来、カメラ位置が既知である任意の視点から得た複数枚の色画像を用いて、物体の三次元形状を Voxel 集団として復元する手法である。この手法を利用する際、Carving を行うための初期形状をどのように求めるかが課題となる。通常は、対象物と区別しやすい背景のもとで画像を取得し、輪郭に基づく制約（シルエット制約）を利用して初期形状を生成することが多いが、本研究では、ロボットが移動するのは実環境であり、取得する画像から対象物の輪郭のみを抽出することは困難である。

一方、本研究においては、カメラ姿勢と物体の疎な形状の復元により、視点の異なる複数枚の画像が取得できており、それ

らを撮影したカメラの姿勢も既知である。そこで、voxel の初期形状をアフィン不変量を利用して生成する。その結果を Space Carving に入力することで、全体の処理を高速化する。

6.2 アフィン不変量を利用した密な形状復元

カメラ姿勢が既知であれば、2枚の画像間で各画素の対応を取ることで密な三次元形状を復元できる。本研究では、カメラ姿勢に加え特徴点の対応も既知であることから、密な形状復元を高速に行う手法を提案する。

ある画像内において指定した画素に対して、近傍に存在し、別の画像に追跡できている特徴点3点から、その画素を内包するような三角形を構成する。本研究では、二次元ドロネー分割を利用して三角形領域を生成する。この三角形は小領域となるため、領域内の画素の対応はアフィン変換で近似できる。具体的には、以下の式によりアフィン不変量を算出し、異なる画像間で画素の対応を付ける。不変量とは、あるモデルの変形前と変形後において、その間の変形の変換に対し不変な表現を意味する。

$$\mathbf{z} = \alpha(\mathbf{p}_2 - \mathbf{p}_1) + \beta(\mathbf{p}_3 - \mathbf{p}_1) + \mathbf{p}_1 \quad (5)$$

ここで、 $\mathbf{z}$ は指定した画素、 $\mathbf{p}_n (n = 1, 2, 3)$ は、画像内の特徴点であり、 α, β はアフィン不変量を表す。式 (4) から、 α, β が算出できる。一方、画像 A 中の $\mathbf{z}$ に対する画像 B 中の推定座標 $\mathbf{z}'$ は、以下の式で表される。

$$\mathbf{z}' = \alpha(\mathbf{p}'_2 - \mathbf{p}'_1) + \beta(\mathbf{p}'_3 - \mathbf{p}'_1) + \mathbf{p}'_1 \quad (6)$$

$\mathbf{p}'_n (n = 1, 2, 3)$ は、画像 A 中の特徴点に対応付けられた画像 B 上の特徴点を表す。式 (5) 式では、右辺のパラメータはすべて既知となるため、 $\mathbf{z}'$ が一意に求まる。

この枠組みを利用することで、現在の画素 (Fig. 5, P) が別の画像上のどこに移動したかを求めることができる (Fig. 5, Q)。対応がついた画素の三次元位置は、モーションステレオと同様の計算により求まる。この手法は、画像特徴点の対応を利用するため、相関による画素の対応付けなしに高速に密な復元が可能である。

ただし、本手法ではアフィン変換で三角パッチの対応をとっているため、画素の推定位置には近似による誤差が生じる。そこで、復元する画素を選択することにより、誤差の大きな点が復元されることを制限する。具体的には、以下の条件に当てはまる場合は復元処理を行わない。

Fig. 5 Affine invariant

- 三角パッチの頂点間の三次元位置が、カメラの奥行き方向に大きく異なる
- 三角パッチの面積が所定の閾値よりも大きい
- 推定した画素 Q の座標が、もとの画素 P の座標によるエッジ極線から所定の閾値以上の距離にある
- 画像 A から推定した画像 B の画素の輝度値が、実際の画像 B の画素の輝度値と大きく異なる

これらの項目に当てはまる部分については、1 回の復元では結果に空白ができる。しかし、視点の異なる画像間で多数回処理を行うことで、最終的には物体表面がほぼ空白なく覆われる。

6.3 Voxel 空間への統合

前節の手法により密な復元を行った結果を、Voxel 空間に統合する。アフィン不変量を利用した復元結果は、二次元アフィン変換の近似により結果に復元誤差が多く含まれる。また、多数枚の画像を用いるため、2 枚の画像ごとに密な形状復元を続けると、冗長なデータが非常に多くなる。

そこで、6.2 節の手法により 2 枚の画像ごとに復元した結果を Voxel 空間に統合する。Voxel 空間については、これまでに判明した物体の位置と大きさを考慮して決定してやればよい。

具体的には、定義した Voxel 空間において、2 枚の画像から復元した画素の三次元位置を含んでいる Voxel に対し投票する。ただし、6.2 節の復元では物体の曲面部分などの詳細な形状が正確に表現できない可能性がある。そこで、近傍の Voxel にも投票することで、次節で述べる Space Carving の初期形状として不足がないよう Voxel を補間する。他の画像間でも同様の処理を行い、最終的な投票値が所定の閾値を超えた Voxel のみを残す。復元精度の低い外れ値は、この閾値処理により除去できる。これより得られる復元結果は、形状に厚みを持った Voxel の集団として表される。

6.4 Space Carving による形状の補正

上記の処理で得られた Voxel 集団を初期形状とし、Photometric な情報に従い削る。具体的には、Fig. 6 に示すように、ある Voxel V が画像 A 上に投影された際、その投影像 v_a に含まれる画素の輝度値を記録しておく。そして別の画像 B 上にその Voxel V が投影された際その投影像 v_b の画像領域に含まれる輝度値を調べ、画像 A の v_a の輝度値と異なればその Voxel は消去する。これを複数枚の画像間で繰り返し、最終的な対象

物の形状を求める。

Space Carving は幾何学的な近似を含まず、画像の Photometric な情報により処理を行うためアフィン不変量を利用した密な形状復元の誤差を補正できる。また、冗長なデータを削減し、厚みのないサーフェスモデルを構築できる。

7. 実験

7.1 実験方法

2 輪駆動型の移動ロボット「山彦」に、Canon VC-C4 パンチルトカメラを搭載したシステムを構築した。画像取得には、HI-TACHI 製の画像処理モジュール「IP7500」を用いた。画像の解像度は 512×440 pixel である。画像列に対する処理は、Pentium4, CPU2.8 GHz の計算機を用いてオフラインで行った。

床面上に設置した対象物に対し物体から 1,500 [mm] 離れた位置にロボットを設置した。対象物 (Fig. 7) は、直径 220 [mm] のサッカーボールである。ロボットには、物体に向かって走行させた後、その周囲に半径 800 [mm] の円経路を与えた。

画像はロボットが 20 [mm] 進むごとに取得した。この理由は、画像間で視差が大きな場合やスケールの変化が大きい場合に KLT-Tracker が追跡ミスを起こしやすいことから、視差を十分小さくするためである。ロボットの動作開始とともに対象物の発見処理 [16] を行った。ロボットから離れている背景部分を除去し、物体とロボットの相対位置、復元のスケールを算出した。それらを初期値として、物体の形状モデリングを行った。

7.2 復元結果

対象物の周囲より取得した画像から、カメラ姿勢と物体の疎な形状を復元した。特徴点の三次元復元の結果を Fig. 8 に示す。ここでは、1,300 点の特徴点の三次元位置と、190 箇所のカメラの姿勢を復元した。初期復元フェーズにおける因子分解法は、20 [mm] ごとに取得した複数枚の画像を用いて行った。一方で、モーションステレオには視差が必要なことから、逐次復元フェーズでの三次元復元は取得間隔が 60 [mm] 以上離れている 2 枚の画像間で行った。

次に、得られたカメラ姿勢と、画像特徴点の推移を用いて密な復元を行った。Voxel のサイズは一辺 2 [mm] とし、Voxel 空間の解像度は $120 \times 120 \times 120$ とした。Voxel サイズの決定方法は以下のとおりである。本実験では、カメラと物体間の距離と、

Fig. 6 Space carving

Fig. 7 Target object

Fig. 8 3D feature points

Fig. 9 Dense reconstruction from two images

カメラの内部パラメータの関係から、画像上の1画素は対象物付近で約1[mm]の大きさに相当する。そこで、画素単位の誤差が影響せず、かつ小さなサイズとしてVoxelの一辺を2[mm]に決定した。

アフィン不変量を利用した復元は、画像列全体で80ペアを行った。Fig. 9は、ある画像ペアでのアフィン不変量を利用した復元結果である。処理の過程では、画像ひとペアにつき平均2万7,000点の画素の三次元位置が求められた。処理全体で得られるデータ量は、この80倍となった。ここでの問題として、Fig. 9に示すように、復元結果の表面に誤差を持つ点が散らばったように復元されることがあった。しかし、この処理は異なる視点の画像により多数回行われるため、これらの外れ点は、Voxelの投票値を閾値処理する段階で解消された。

Voxel空間へ統合した結果を初期値とし、Space Carvingを行った。Fig. 10(1)は、Space Carvingで得られた最終的な形状モデルである。この結果では、床と対象物の一部が合体しているが、これについては次節の方法で対処する。Space Carvingの初期形状の生成では、Votingの結果、投票値が50以上となったVoxelのみを選択した。Voting後のVoxel数は約20万個であったが、Space Carvingを行った結果、最終的に残ったVoxel数は約4万個であった。

Fig. 12は、提案手法により生成したグローブの復元結果である。

7.3 床面の除去

Fig. 10(1)では、対象物の他にも物体領域が残るという問題

Fig. 10 Final 3D result

があった。この原因は、アフィン不変量を利用した復元に、床面の傷などから抽出された特徴点が含まれていたためであった。対象物上の画像特徴点と床面上の画像特徴点との間で三角パッチが生成され、Voxel空間に復元結果が投票された。

そこで、物体は床面上に置かれているという前提を利用し、特徴点の三次元位置のz座標を調べることで、床面を除去する処理を加えた。具体的には、復元した特徴点が床面と同じ高さであれば、その点を以降の処理に利用しないとした。その結果、Fig. 10(2)に示すように、良好な結果が得られた。

この処理の有効性を確認するため、床面に多くのテクスチャがある場合にもモデリングの実験を行った。具体的には、Fig. 11(1)に示すように、床面に模様のある紙を設置した。この結果をFig. 11(2), (3)に示す。他の例と同程度の復元精度を保ちながら、物体領域を切り出した形でモデリングできた。ただし、対象物に非常に近接して他の物体が存在する場合は、一般に極めて難しい問題であり今後の課題である。

7.4 評価

7.4.1 処理時間

疎な復元の処理時間の内訳は、以下のとおりである。特徴点の抽出・追跡については、画像1枚につき400[ms]であった。因子分解法と非線形最小化は25枚の画像で行い、2.5[s]を要した。また、以後のカメラ姿勢と物体の疎な形状算出には画像1枚につき40[ms]を要した。すなわち、初期値生成後は、画像1枚につき500[ms]以内で、物体の疎な形状が得られる。

一方、密な復元の処理時間の内訳は、以下のとおりである。アフィン不変量を利用した復元の処理時間は、一つのペアに

(1) Original image

(2) 3D position of feature points

(3) Voxelized model

Fig. 11 Experiment 2

Original Image

3D Modeling Result

Fig. 12 Modeling example

160 [ms], 全体では 13 [s] を要した. Voxel への統合と Space Carving は全周囲の画像群で行い, 処理時間は 90 [s] であった.

7.4.2 精度

Fig. 10 (2) のサッカーボールの直径を実測値と比較したところ, これらの誤差は 10 [mm] 以内で納まることを確認した. また, Fig. 12 のグローブの例では, モデリング結果の高さ, 縦幅, 横幅を実測値と比較したところ, これらの誤差はそれぞれ約 10 [mm] であった.

Space Carving によって厚みが減る様子を Fig. 13 に示す. この図は, 床面と平行な平面で, ボールの復元結果の中心を切断したものである. Voxel 空間に投票した結果では, 復元形状は最大 20 [mm] 程度の厚さを持った状態で得られている. 外側の Voxel から測定した復元形状の直径は, 230 [mm] を超えて

いるところが多い. これに対し Space Carving を行った結果, Fig. 13 の中央の拡大図のように厚みのないサーフェスモデルが得られた. 直径についても, 真値である 220 [mm] に近い値となった.

7.4.3 考察

上述のように, Space Carving によって得た最終的な形状では, 目的の復元精度を達成できた.

疎な復元では, ロボットの走行中に適用可能な速度で, カメラ姿勢と物体形状が得られた. ただし, ここでは画像特徴点の追跡に多くの処理時間を要しており, この部分の高速化が今後の課題である.

密な復元では, アフィン不変量を利用した復元により, Space

Fig. 13 Voxelized model

Carving の初期形状を高速に生成できた。Voxel 空間への統合後の結果は、20 [mm] 程度の厚みを持つ状態で得られた。このとき、厚みの中に形状の真値が含まれていたことを考慮すると、Voxel 空間への統合後の結果はそれほど悪くはないと言える。一方で、Space Carving は処理時間を要した。

これより、Space Carving を行うかどうかは目的に応じて決めることがよいと考えられる。例えば、形状が単純な物体の場合や、物体ハンドリングを迅速に行いたい場合には、Space Carving を行わず、Voting 後の厚みのあるモデルに対して、その厚みの中央値を取った近似形状を利用することも考えられる。一方で、慎重に扱わなければならない物体などは、把持計画の際に正確なモデルを持つ必要がある。また、獲得した物体モデルを後で再利用するためにデータベースに登録する際なども、なるべく高精度でモデルを生成する必要がある。このような場合には、Space Carving も処理に含め、できるだけ高い精度でモデリングを行うのがよいであろう。

8. ま と め

本稿では、移動ロボットに搭載するための物体の形状モデリング手法を提案した。また、実際に移動ロボットを走行させながら取得した画像列を用いて、実画像による実験から三次元形状モデリングを行い提案手法の有効性を示した。

今後の課題として、形状モデリングに最適となるようなロボットの視点移動計画がある。また、処理を高速化し、移動ロボットの自律的な形状モデリングの実現を目指す。

参 考 文 献

- [1] C. Tomasi and T. Kanade: "Shape and Motion from Image Streams under Orthography: a Factorization Method," IJCV, vol.9, no.2, pp.137-154, 1992.
- [2] R. Hartley and A. Zisserman: Multiple View Geometry in Computer Vision. Cambridge University Press, 2000.
- [3] 藤木, 蔵田: "透視射影に基づく Paraperspective モデルの逐次型因子分解法", 第 5 回画像の認識理解シンポジウム (MIRU2000), vol.1, pp.403-408, 2000.
- [4] 徐: 写真から作る 3 次元 CG—イメージ・ベースド・モデリング&レンダリング—. 7 章, 8 章, 近代科学社, 2002.
- [5] J. Shi and C. Tomasi: "Good Features to Track," Proc. IEEE CVPR, pp.593-600, 1994.
- [6] J. Miura, Y. Shirai and N. Shimada: "Development of a Personal Service Robot with User-Friendly Interfaces," 4th Int. Conf. Field and Service Robotics, pp.293-298, 2003.
- [7] K. Kutulakos and S. Seitz: "A Theory of Shape by Space Carving," IJCV, vol.38, no.3, pp.198-218, 2000.
- [8] L. Iocchi, D. Mastrantuono and D. Nardi: "Visually Realistic Mapping of a Planar Environment with Stereo," Int. Symp. on Experimental Robotics (ISER), pp.521-532, 2000.
- [9] M. Okutomi and T. Kanade: "A Multiple-Baseline Stereo," IEEE Trans. on PAMI, vol.15, no.4, pp.353-363, 1993.
- [10] M.A. Fischler and R.C. Bolles: "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," Comm. of the ACM, vol.24, pp.381-395, 1981.
- [11] M. Tomono and S. Yuta: "Indoor Navigation based on an Inaccurate Map using Object Recognition," Proc. of IROS2002, pp.619-624, 2002.
- [12] C.J. Poelman and T. Kanade: "A Paraperspective Factorization Method Shape and Motion Recovery," Tech. report CMU-CS-93-219, Computer Science Department, Carnegie Mellon University, 1993.
- [13] M. Sainz, N. Bagherzadeh and A. Susin: "Carving 3D Models from Uncalibrated Views," CGIM 2002, pp.144-149, 2002.
- [14] T. Sato, M. Kanbara, N. Yokoya and H. Takemura: "Dense 3-D Reconstruction of an Outdoor Scene by Handhelds-baseline Stereo using a Hand-held Video Camera," IJCV, vol.47, no.1-3, 2002.
- [15] S. Thurn, W. Burgard and D. Fox: "A Probabilistic Approach to Concurrent Mapping and Localization for a Mobile Robot," Machine Learning and Autonomous Robots, vol.31, no.5, pp.1-25, 1998.
- [16] 山崎, 友納, 坪内, 油田: "移動マニピュレータによる物体ひろいあげに関する研究", 第 8 回ロボティクス・シンポジウム予稿集, pp.337-342, 2003.
- [17] 山崎, 友納, 坪内, 油田: "移動ロボットによる物体拾い上げのための対象物の形状獲得—ロボаст性と復元精度の向上—", 第 4 回 SICE システム・インテグレーション部門講演会, 2003.
- [18] 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," Artificial Intelligence Journal, vol.78, pp.87-119, 1995.

山崎公俊 (Kimitoshi Yamazaki)

1979年9月2日生。2002年筑波大学工学システム学類卒業。同年同大学大学院システム情報工学研究科入学。現在に至る。自律移動ロボットの視覚と動作計画に関する研究に従事。IEEE 会員。

(日本ロボット学会学生会員)

坪内孝司 (Takashi Tsubouchi)

1959年8月7日生。1988年筑波大学大学院博士課程工学研究科修了。宇都宮大学工学部情報工学科助手、東京大学工学部計数工学科助手を経て、1994年筑波大学電子・情報工学系講師。現在、同大学大学院システム情報工学研究科知能機能システム専攻助教授。移動ロボットのシステム開発に関する研究に従事。日本機械学会、人工知能学会、IEEE の会員。

(日本ロボット学会正会員)

友納正裕 (Masahiro Tomono)

1960年5月10日生。1985年東京大学大学院修士課程修了(電子工学専攻)。同年 NEC 入社。1999年退社。2002年筑波大学大学院博士課程修了。工学博士。現在、科学技術振興機構さきがけ研究者。ロボットの環境認識・モデリング、行動制御、ソフトウェアアーキテクチャなどに興味を持つ。情報処

理学会会員。

(日本ロボット学会正会員)

油田信一 (Shin'ichi Yuta)

1948年3月23日生。1975年慶應義塾大学大学院(電気工学専攻)修了。現在、筑波大学副学長。知能移動ロボット、ロボットセンサ、ロボット用コントローラなどの研究に従事。自動制御や信号処理の理論と応用に興味を持つ。工学博士、IEEE、計測自動制御学会などの会員。(日本ロボット学会正会員)