

カメラを搭載した移動ロボットによる物体のオンライン3次元モデリング

山崎 公俊 (筑波大) 友納 正裕 (筑波大) 坪内 孝司 (筑波大) 油田 信一 (筑波大)

Online 3-D Object Modeling by a Mobile Robot with a Camera

*Kimitoshi Yamazaki, Masahiro Tomono, Takashi Tsubouchi and Shin'ichi Yuta
(University of Tsukuba)

Abstract— This paper describes how to acquire 3-D object shape by a mobile robot at online. A single camera is placed on the robot, images of the target object are captured in streams by the camera on a mobile robot moving around the object. Computer vision techniques are utilized under the condition that shape, size and accurate position information of the target object are not given in advance, but the robot finds the target object and reconstructs the 3-D object shape by itself. Experimental results show the good result and prove our 3-D modeling system on a mobile robot in the point of robustness and accuracy.

Key Words: 3-D Modeling, Vision, Mobile Robot, Single Camera, Mobile Manipulator

1. はじめに

本論文は、物体の3次元形状モデルを生成する移動ロボットシステムに関して述べるものである。本システムでは、外界センサとして単眼のカメラを搭載し、物体の周囲を走行しつつ複数枚の画像を取得する。得られた画像列を利用して、ロボットの走行中に、物体の形状モデルを自律的に生成する。

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欠かせない重要な要素である。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 [3] などが、成果として上げられる。

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。ロボットの行動に必要な全ての物体のモデルをあらかじめ与えておくことは、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自律的に生成できれば、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、移動ロボットが自律的に、実環境に存在する物体の3次元形状モデルを生成する機能を開発し、報告をおこなってきた [5]。そこでは、対象とする物体の形状・大きさは未知とし、マーカなどを一切与えない状態で、物体の密な形状を獲得できた。本稿では、提案した復元手法の移動ロボットへの統合について述べる。また、オンライン化で発生した問題への対処と、モデリング処理の高速化の手法を提案する。実際にオンラインで実験をおこない、形状モデリングをおこなった結果について述べる。

2. 問題設定とアプローチ

実験機として、車輪型移動ロボット「山彦」(Fig.1)に、単眼のパンチルトカメラを搭載したシステムを用いる。ロボット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、自己位置の推定と、物体の形状獲得をおこなう。なお本稿では、カメラの位置と方位を合わせてカメラ姿勢と呼ぶ。



Fig.1 車輪型移動ロボット

2.1 対象物の条件

ロボットは、物体が複数存在する環境を走行しながら、注視する対象物を特定する。対象物に関しては、以下の仮定をおく。

1. 対象物の形状に関する事前知識は持たせない
2. 対象物には人工的なマーカを付けない
3. 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

対象物の形状は未知とするが、平面だけでなく、曲面を含んだ物体も対象とする。また、物体に対しマーカ等の付加情報は一切与えない。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、困難な問題設定である。

3. の仮定は、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。本研究においては、特徴点の抽出・追跡処理には KLT-Tracker [2] を利用する。

2.2 処理の要件

本研究での処理の要件を、以下のように定める。

- 注視する対象物をロボットが自律的に決定する。
- 逐次的な形状獲得をおこなう。

本研究では、ロボットは実環境で動作するため、対象物以外にも複数の物体が存在することが想定される。

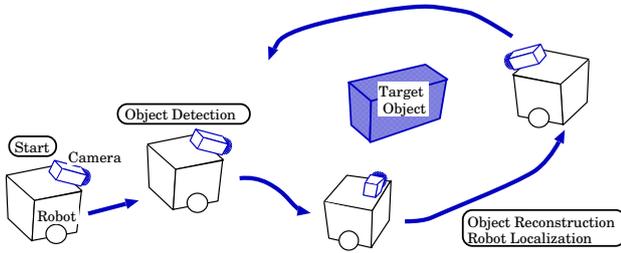


Fig.2 処理の流れ

すなわちロボットには、実環境を観察し、注視のための対象物を自律的に決定する能力が求められる。

また、対象物の大きさ・形状は未知であるため、ロボットには、ある視点から得られた物体の形状を元に、次の視点を動的に決定することが求められる。そのためには、ロボットが走行しながら得た画像を即座に処理し、そのとき観察している物体の部分形状を獲得できるような、逐次的な処理が望まれる。

これらの要件を満たす処理を、ロボットの走行中に実行可能にする。

2.3 移動ロボットにおける課題

本研究で対象としている小型物体の形状獲得には、高精度でカメラの位置・姿勢を知る必要がある。一方、筆者らの用いている移動ロボットにおいて、搭載したカメラの姿勢を直接的に推定する手段は、オドメトリ(車輪の回転数から測位するロボットの自己位置)と、パン・チルトカメラの角度情報である。しかし、これらから推定するカメラ姿勢には、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、カメラの角度の誤差などの要因により、誤差が含まれる。

そこで、ロボットの動作が単純で、直接的に推定できるカメラ姿勢が真値に近いと想定される場合、誤差を補正することでカメラの姿勢を推定する。それ以外の場合は、画像情報から、カメラの姿勢と物体の形状を推定する。本研究においては、環境中を観察し対象物を特定する処理は、直進走行による単純な動作でおこなうものとする。対象物を注視し詳細な形状を獲得する処理では、対象物の周囲を走行しパン・チルト角を同時に回転させるような複雑な動作をおこなう。

3. 物体の形状モデリング

形状モデリングアルゴリズムを説明する。詳細は、文献 [5] で説明している。

3.1 アルゴリズムの構成

ロボットの動作を、1. 床面上を探索しながら直進走行、2. 特定された対象物へ近づく走行、3. 対象物を注視しながらその周囲を走行、の3フェーズに分ける。各フェーズにおいて、ロボットは以下の処理をおこなう。

1. (対象物の特定)
床面に在る複数の物体から、注視の対象物を特定
2. (カメラ姿勢と形状の初期値生成)
対象物に接近しながら取得した複数枚の画像から、カメラ姿勢と物体形状の初期値を生成
3. (カメラ姿勢と形状の同時推定)
対象物の周囲を走行しながら画像を1枚取得するごとに、カメラ姿勢と物体形状を逐次推定

3つに分けたそれぞれの状況に応じたアルゴリズムを構築し、高速・高精度に形状を獲得する。

3.2 形状復元手法

本研究では、3種類の既存の形状復元手法を利用する。モーショステレオ、因子分解法、非線形最小化である。これらの手法の特性を十分に生かすよう、状況に応じて手法を使い分ける。

モーショステレオでは、カメラ姿勢が既知の場合に、以下の関数を線形最小化し、特徴点の3次元位置を求めることができる [1]。カメラ姿勢をあらかじめ求めておく必要があるが、処理は高速である。

$$C = \|\mathbf{X} - s_1 \tilde{\mathbf{m}}_1\|^2 + \|\mathbf{X} - s_2 \mathbf{R} \tilde{\mathbf{m}}_2 + \mathbf{T}\|^2 \quad (1)$$

$\tilde{\mathbf{m}}_1$ 、 $\tilde{\mathbf{m}}_2$ 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3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。

因子分解法 [4] では、画像から得る特徴点の推移情報から、カメラ姿勢と物体形状を同時に算出できる。解は線形近似の誤差を含むが、処理が高速に実行できる。手順としては、複数枚の画像全てで観察される複数個の特徴点推移を並べた行列 \mathbf{W} を作成する。この行列のある条件の元で因子分解し、カメラの姿勢行列 \mathbf{M} と、物体の形状行列 \mathbf{S} を同時に求める。

$$\mathbf{W} = \mathbf{M}\mathbf{S} \quad (2)$$

行列 \mathbf{M} はカメラの運動を表しており、行列 \mathbf{S} は特徴点の3次元位置を表している。

非線形最小化では、特徴点の3次元位置やカメラ姿勢の誤差を同時に補正できる。この手法は、最適化したい変数の初期値を必要とする。繰り返しによる最適化処理をおこなうため、処理時間はかかる。ただし、変数の数を制限したり、初期値が十分真値に近い場合には、高速に動作する。

$$C = \sum_{j=0}^2 \sum_{i=0}^P \left(\frac{\mathbf{r}_{xj}^T \mathbf{m}_{ij}}{\mathbf{r}_{zj}^T \mathbf{m}_{ij}} - \frac{X_i + t_{xj}}{Z_i + t_{zj}} \right)^2 + \sum_{j=0}^2 \sum_{i=0}^P \left(\frac{\mathbf{r}_{yj}^T \mathbf{m}_{ij}}{\mathbf{r}_{zj}^T \mathbf{m}_{ij}} - \frac{Y_i + t_{yj}}{Z_i + t_{zj}} \right)^2 \quad (3)$$

\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3次元座標を表す。

3.3 形状復元アルゴリズム

3つのフェーズそれぞれについて、ロボットの画像取得状況と、それに応じた形状復元手法の組み合わせ方について述べる。

3.3.1 対象物の特定

注視する対象物を特定し、ロボットと対象物の正確な位置関係を得る。

最初のフェーズである対象物の特定処理では、ロボットには単純な直線経路を与える。このとき走行距離が短ければ、オドメトリの値は誤差が少ないため、これらの値をカメラ姿勢の推定に利用できる。形状復元の手法としては、モーショステレオを利用する。この手法は、カメラ姿勢が既知の基で、高速に3次元形状が得られる。さらに、この復元結果を初期値として、非線形最小化によりカメラ姿勢と物体形状を補正する。最小化の変数の数については、モーショステレオの結

果を基に復元精度の良い特徴点を選び出して非線形関数に入力することで、高速な処理を実現できる。

この処理により床面上の物体の有無を調べ、復元結果をクラスタリングし対象物を特定する [6]。

3.3.2 カメラ姿勢と形状の初期値生成

カメラ姿勢と形状の初期値生成は、対象物を特定した後、ロボットが物体へ接近する間におこなわれる。ここでは、ロボットから観察される対象物の姿勢変化が少なく、画像間で特徴点の入れ替わりが少ない。また、この動作ではオドメトリ等から推定されるカメラ姿勢には誤差が多く含まれるようになる。そこで、因子分解法を利用する。因子分解法は、画像情報のみからカメラ姿勢と物体形状を同時に獲得できるため、本フェーズでの状況に適している。この手法から得た解に含まれる線形近似の誤差に対しては、非線形最小化を適用し、補正をおこなう。

以上より、注視する対象物の形状と、それを観察しているカメラ姿勢を高精度に獲得し、以降の同時推定処理の初期値を生成する。

3.3.3 カメラ姿勢と形状の同時推定

このフェーズでは、ロボットは物体の周囲を走行する。ここでは、ロボットから観察される物体の姿勢が随時変化していくため、画像から抽出できる特徴点は頻繁に入れ替わる。そこで、カメラの姿勢と物体の形状を逐次的に獲得する。具体的には、新しく画像を取得するたびに以下の処理をおこなう。

- 3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推定
- 求めたカメラの姿勢から、新たに抽出された特徴点の3次元位置を算出

新たに画像を取得するたびに非線形最小化をおこない、カメラ姿勢を逐次得る。非線形最小化の変数はカメラ姿勢のパラメータのみであるため、処理は高速である。

一方で、推定されたカメラ姿勢を用いて、モーションステレオをおこなう。特徴点の3次元位置を一点ずつ求める。このとき、3次元位置が正確に求まらない特徴点も存在するため、復元結果を現在の画像上に逆投影し、トラックによる追跡結果との差分を測ることで復元精度を評価する。十分な精度を持つと判断された特徴点のみを、以降で3次元位置が既知の点とし、カメラ姿勢の算出に利用する。

以上より、画像を取得するたびにカメラの姿勢と対象物の形状を逐次的に求める。

4. オンライン化における問題と対処

3章の流れを、移動ロボットがオンラインで実行できるよう実装した。その際、カメラ姿勢と形状の同時推定処理において、以下のような問題があった。

- ロボットやカメラの回転時に画像を取得すると、ぶれなどの要因で多数の特徴点が追跡不可能になる。
- 追跡ミスや追跡精度の低い特徴点が多いため、カメラ姿勢の推定に失敗したり、誤差を含む点の除去に処理時間を要する。

これらの問題に対処するため、提案手法に改良を加えた。以下にその詳細を説明する。

4.1 取得状態の悪い画像の検出と破棄

ロボットの走行やカメラの動作は常に安定しているとは限らず、画像取得のタイミングによっては、ぶれた画像が取得される事態が起こりうる。このような画

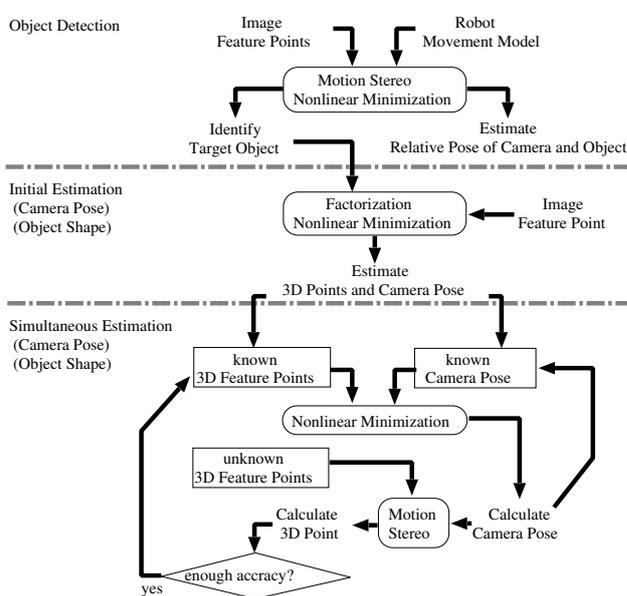


Fig.3 アルゴリズム

像では、多くの特徴点が追跡不可能となり、追跡精度も全体的に低下するため、カメラの姿勢を推定できずに処理が破綻する恐れがある。

対処として、 $n-1$ 枚目の画像から n 枚目の画像へ特徴点追跡をおこなった際、所定の個数以上の特徴点が追跡不可能になったら、 n 枚目の画像を破棄し、 $n-1$ 枚目と $n+1$ 枚目の画像間で特徴点を追跡する。この処理により、特徴点の多くが追跡可能な画像のみを選択することで、ロボットの自律処理を実現する。

4.2 RANSAC の応用

ロボット上のカメラから得る画像列は、動画ほどの連続性を有していないため、トラックによる特徴点追跡処理において、追跡ミスや精度の低下が頻繁に起こる。これらの特徴点からカメラ姿勢を推定することは、処理の破綻につながる。

これに対し筆者らは、RANSAC の枠組みを用いた対処法を適用し、誤対応・低精度の特徴点を取り除いた [7]。提案手法では、ある画像内で観察され、既に3次元位置が判っている特徴点のうち6割程度をランダムに選択し、それらからカメラ姿勢を推定することを、数百回程度繰り返す。全ての試行の中で一番評価が良い値を、カメラ姿勢の推定値とする。

この手法では、サンプリングの候補とする特徴点集団に外れ値が少なければ、サンプリングの回数を減らすことができ、処理を高速化できる。そこで、新たに設定した以下のルールにより特徴点をフィルタリングし、処理を高速化する。

- 一つ前のカメラ姿勢を推定した際、その画像内に存在する既知の特徴点を以下の式で評価する。 eva の値が閾値以下の点のみ、現在のカメラ姿勢におけるサンプリング候補の対象とする。
- 現在のカメラ姿勢を RANSAC の枠組みで評価する際、下式の値が所定の閾値以下であれば、良い精度でカメラ姿勢が求まったとし、処理を終了する。

$$eva = \frac{1}{P} \left\{ \sum_{i=0}^P \left(\frac{\mathbf{r}_x^T \mathbf{m}_i}{\mathbf{r}_z^T \mathbf{m}_i} - \frac{X_i + t_x}{Z_i + t_z} \right)^2 + \sum_{i=0}^P \left(\frac{\mathbf{r}_y^T \mathbf{m}_i}{\mathbf{r}_z^T \mathbf{m}_i} - \frac{Y_i + t_y}{Z_i + t_z} \right)^2 \right\}$$



Fig.4 原画像

5. 実験

自律移動ロボットに構築したシステムを用いて実験をおこなった。画像は、HITACHI 製の画像処理モジュール「IP7500」により、 512×440 pixelのサイズで、ロボットが20mm 走行するたびに取得し、ノートPCへ転送した。ノートPC(Pentium M 1.7GHz)では、特徴点の抽出・追跡、カメラ姿勢の推定と形状モデリングをおこなった。

床面には複数の物体を置き、ロボットに一番近い物体を対象物とするよう、あらかじめ命令を与えた。対象とする物体は、直径220mmの球体 (Fig.4) である。

対象物の特定処理は、処理開始から取得した10枚の画像から特徴点を抽出・追跡し、最初と最後の画像を用いておこなった。処理時間については、1枚の画像からの特徴点抽出・追跡に450msec、モーションステレオ・非線形最小化の処理に30msec、クラスタリング処理に20msecを要した。

カメラ姿勢と物体形状の初期値生成処理では、特定した対象物に近づきつつ取得した25枚の画像を用いておこなった。処理時間については、25枚分の特徴点情報をを用いて因子分解法・非線形最小化を実行するのに、約5secを要した。

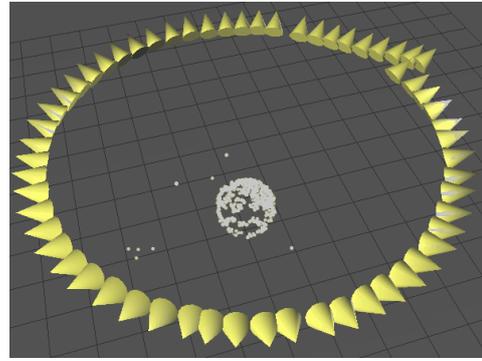
カメラ姿勢と物体形状の同時推定処理では、204枚の画像に対し、新しく画像を取得する度に処理をおこなった。特徴点追跡に追加抽出の処理を加えて850msec、カメラ姿勢と特徴点の3次元位置を求める処理では、画像1枚あたりの平均時間が80msecの処理時間を要した。

最終的に、229ヶ所のカメラ姿勢と、1800点程度の特徴点の3次元位置を得た。Fig.5 上に、カメラ姿勢と特徴点の3次元位置の結果を示す。ここでは、円錐の底辺がカメラの向きを表しており、物体を観察しながら移動するカメラの軌跡が復元されている。Fig.5 下は、文献[5]の手法を用いて得た密な復元の結果である。密な復元については、オフラインで処理をおこなった。復元精度については、多少歪んでいるものの、形状・大きさの誤差は最大で10mm以内に納まっていることを確認した。また、その他の例として、グローブを復元した例でも、同程度の復元精度があることを確認した。

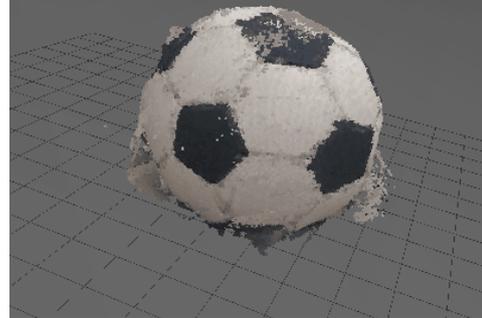
6. まとめ

本稿では、[5]で報告した形状モデリングアルゴリズムの移動ロボット上への統合に関して述べた。実際の移動ロボットを用いて、対象物を特定し、その正確な形状を獲得するまでの一貫した動作を実現した。

今後の課題として、獲得した形状からカメラの視点とロボットの走行経路を動的に決定することがある。



Socker Ball



Grub



Fig.5 復元結果 (上段: カメラ姿勢と疎な形状、中段: 密な形状、下段: グローブのモデリング結果)

参考文献

- 1) 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.
- 2) J. Shi and C. Tomasi: "Good Features to Track", CVPR, 1994.
- 3) J. Miura, Y. Shirai and N. Shimada: "Development of a Personal Service Robot with User-Friendly Interfaces", 4th Int. Conf. on Field and Service Robotics, pp.293-298, 2003.
- 4) Poelman C.J and Kanade T.: "A Paraperspective Factorization Method Shape and Motion Recovery" tech. report CMU-CS-93-219, Computer Science Department, Carnegie Mellon University, 1993.
- 5) K.Yamazaki, M.Tomono, T.Tsubouchi and S.Yuta: "3-D Object Modeling by a Camera Equipped on a Mobile Robot" ICRA, 2004.
- 6) 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ア, 2003.
- 7) 山崎, 友納, 坪内, 油田: 「移動ロボットによる物体ひろいあげのための対象物の3次元形状獲得」SI2003, 2003.