

拡張現実を用いた船殻認識システム

学生員 宮本達也* 正員 木村元**

Hull recognition system using augmented reality

by Tatsuya Miyamoto, *Student Member* Hajime Kimura, *Member*

Key Words: Augmented Reality, PTAM, Web Camera

1. 緒言

どのような生産ラインにも言えることだが、造船においても作業工程の「遅滞」は極力減らさなければならない。作業の「遅滞」が大きくなれば、収益や信用の低下が危ぶまれるからである。そのため、どの工程において「遅滞」が生じているのかという情報が必要となる。しかし現在の造船現場において、進行状況の報告は日・週単位であり、さらにその作業は現場作業責任者の仕事となっている。より詳細な報告を必要とした場合、作業責任者がその作業に追われ更に作業効率が低下し「遅滞」が生じてしまう可能性がある。

この問題の解決案の一つとして、自動で瞬時に進行状況を把握するシステムが存在すれば良いと考えられる。このようなシステムの一つとして 3D 形状認識を用いた CAD モデルとのマッチングを行う手法が考えられる。

3D 形状認識システムには、レーザーレンジファインダを用いた方法や画像処理を用いた方法等が存在し、また 3D 形状認識センサカメラといった専用機器も存在する。しかし、専用機器やレーザーレンジファインダを用いた場合は高コストとなる場合が多く、また既存の画像認識システムの多くは認識環境中にマーカーを配置する必要があった。

しかし近年、の飛躍的なパーソナルコンピュータ (PC) の性能向上に伴い、比較的安価な Web カメラさえあれば AR(拡張現実)を体験できる画像処理技術が登場した。この技術はノート型の PC でも実行できるものであり、場所も制限されないのが特徴である。

このような技術を 3D 形状認識システムに用いることは、システムに「建造進行状況の把握」のみではなく「今後の建造工程の視認的確認」や「施工漏れの検出」等、他の分野での使用の可能性も与える。本論文では、このシステムを構築するに当たって核となる 3D 画像認識と、またその技術の造船現場への適用可能性について考察する。

2. 3D 形状認識システム

3D 形状認識システムとは、その名称の通り認識対象の 3D 形状を把握するために用いられるシステムのことである。その手法としては、レーザーレンジファインダ等の距離センサを用いた手法と、カメラからの画像を用いた手法に大きく分類される。更に、カメラを用いる手法

には 2 つのカメラを用いる方法とカメラを 1 つのみしか用いない方法とに分類することができる。

レーザーレンジファインダ等の距離センサを用いる場合、そのセンサの精度に依存するが、精度の高い形状認識が可能である。しかし多数のセンサを併用でもしない限り、センサを認識対象の形状認識に大きく動かす必要がある。センサ認識範囲は非常に狭いため、今回目的としている船殻のような大型物体では、全体の形状認識に時間がかかる。その点、カメラを使用した場合には一度に広範囲の情報を得ることができる。カメラを 2 台用いれば、広範囲の奥行き情報をより短時間で得ることができる。しかしカメラを 2 台用いた場合、2 台のカメラ間隔を常に維持する必要がある、そのため、これは距離センサを用いた場合にも言えることだが、基本的にセンサの固定が前提となる。

そこでカメラ 1 台を用いる手法がある。この手法は、奥行き変位情報を得るためにカメラを動かし別位置画像を取得することにより、近似的に 2 台のカメラを用いたかのように処理する。この場合必要となるのは、そのカメラが現在どこに存在するのかという情報であり、この情報の取得には認識環境中にマーカーを置くか別途センサを設ける必要がある。

しかし、Klein ら¹⁾が提唱した手法 (以下 PTAM) は、センサとしてカメラを一台しか用いないが、認識環境中にマーカーを使用せず、システム内において特徴点(コーナー)の 3DMap を構築することでカメラ位置情報を推定する。また使用するカメラもある程度の解像度を有していれば、高精度の距離センサや専用の 3D 形状認識センサカメラ等に比べれば比較的安価な Web カメラで事足りるといった優れた特徴を有する。PTAM は商用ではなく研究用のため無料で使用できるが、AR システムとして大変高性能である。また、ソースコードが公開されており、カスタマイズが容易であることから、本研究では PTAM に着目した。

3. PTAM の概要

3.1 主な特徴

PTAM とはカメラ一台を動かすことによりステレオ視を行い、それによって 3D 空間を認識して AR を実行する。その主な特徴は以下の 5 つである。

Tracking スレッドと Mapping スレッドを分割

Mapping をキーフレームに基づいて処理

立体処理(5-point Algorithm²⁾)により初期 3DMap を生成
エピソード探索によって新しい特徴点(角)の初期位置を設定

* 九州大学大学院工学府海洋システム工学専攻

** 九州大学大学院工学研究院海洋システム工学部門

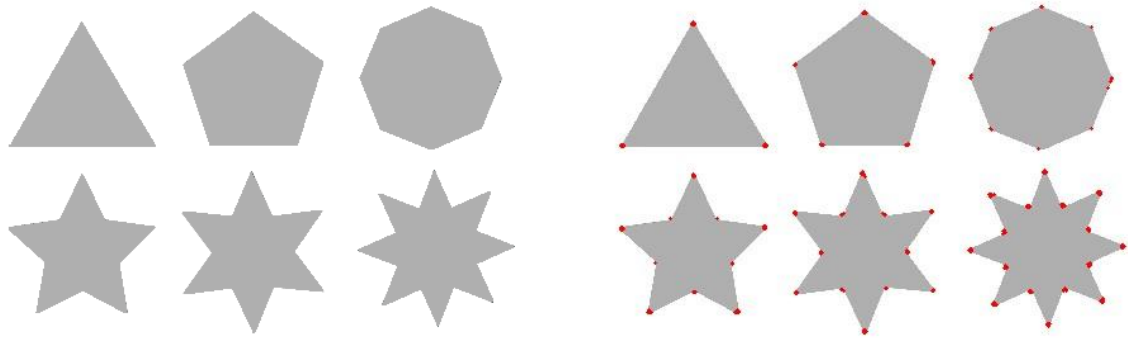


Fig. 1 FAST Demo : Left is an original image, and Right is a image created by FAST

数多くの(数千の)特徴点を 3DMap に表示

また、現在ではほぼ標準である解像度(640*480)さえ確保できてさえいれば、高価な専用機器に比べ安価な Web カメラでも動作する。

3.2 Tracking と Mapping

PTAM でいう Tracking とは「リアルタイムのカメラ姿勢推定を Map に関して保存するシステム」のことを指し、Mapping とは「画像中における特徴点(角)をワールド座標系 (Map 座標系) で保存するシステム」のことを指している。

既存の AR システムの多くでは Tracking と Mapping は同スレッドに存在し、Tracking によって得た特徴点によって Mapping を行い、Mapping によって生成された Map を元に Tracking を行うといった相互関係が成立していた。しかし、この方法はカメラのフレーム毎の計算コストが高くなり、データ連結エラーの原因となっていた。しかし PTAM では、これらのスレッドを分割して処理することによりデータ連結エラーを回避し、さらにはそれぞれのスレッドに様々な利点を与えている。

Tracking スレッドでは、Map 更新という高コスト計算から解放されたことにより計算コストが大幅に削減され、それによって Tracking 性能によりロバスト性のある、coarse-to-fine アプローチを用いている。また Mapping スレッドでは、Tracking によるフレーム毎の計算から解放され、一定時間毎に獲得するキーフレームに基づいた処理が可能となる。これにより、Map 更新はリアルタイム処理という制限に囚われなくなり、さらにバンドル調整を用いることで、よりロバスト性の高い手法を用いることができる。

3.3 特徴点の抽出

特徴点の抽出には FAST³⁾と呼ばれる手法が用いられている。FAST とは画像中の角を検出する手法の一つであり、強化学習を用いることにより処理時間を短縮している。PTAM では FAST によって検出された角を特徴点として処理を行っている。FAST による角抽出の一例を Fig. 1 に示す。左図が元画像であり、右図が FAST 実行後の角抽出位置の色が変更されている画像である。Fig. 1 から、FAST が正確に角を認識していることが分かる。

3.4 実行環境

PTAM は Tracking と Mapping を別スレッドで行うという性質上、CPU は core 2 processor 以上が必須となっている。更に、AR 本来の使用用途であるグラフィックスの表

示には NVIDIA 社製のグラフィックボードが必要とされているが、これは PTAM のサンプルプログラムが NVIDIA 社製のグラフィックボードを用いた環境下で開発されているためであり、自作する場合や変更する場合にはこの限りではない。

また PTAM は Klein らの長期研究により完成したシステムのため、その実行には多数のライブラリが必要である。以下に必要なライブラリと、それが何のライブラリであるのかの簡単な説明を示す。

GLEW⁴⁾: OpenGL 拡張

Lapack and BLAS⁵⁾: Fortran90 数値計算

CMU 1394 camera driver⁶⁾: IEEE1394 カメラ

Pthreads⁷⁾: 並列処理プログラム

Libjpeg for windows⁸⁾: JPEG エンコード/デコード

Fltk⁹⁾: クロスプラットフォーム GUI

Toon: 線形代数

libcvd: 画像処理

gvars3: ランタイム構成とスクリプト

サンプルプログラムは IEEE1394 対応カメラに対して開発されており、Web カメラには対応していない。そのため、Web カメラを用いる場合には別途に以下の二つのライブラリが必要となる。

EWCLIB¹⁰⁾

PTAM for Webcam¹¹⁾

これらのライブラリを組み込むことにより、PTAM を初めて実行することができる。また、PTAM のシステム実行時における各ライブラリを用いた処理の相関図を Fig. 2 に示す。各ライブラリは AR 表示やスレッド管理、画像処理等に大まかに分類することができる。

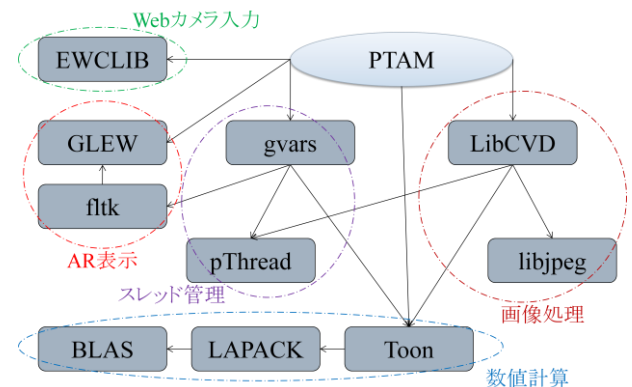


Fig. 2 A correlative figure on PTAM

4. 模型試験

4.1 試験内容

PTAM を用いて作成模型の認識を行い、正確に形状認識できているのか検証を行う。

4.2 模型概要

模型は実スケールの約 1/20 であり、寸法は 450*840*60(mm)である。実際に使用した模型を Fig. 3 に示す。

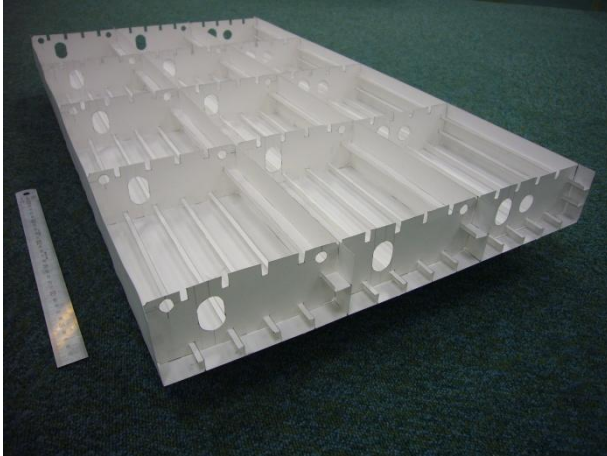


Fig. 3 A 1/20 scale model

4.3 試験結果

Fig. 4 に PTAM で模型を認識した際の実行画面を、Fig. 5 に実行中に生成された 3DMap を示す。各色のドットは認識された特徴点(角)であり、色の違いにより特徴点が認識される coarse-to-fine アプローチでの解像度の相違を示している。また、3 つの線分の集合はキーフレーム毎に保存された推定カメラ位置のカメラ座標であり、格子状の網目は初期 3DMap の基礎平面を示している。

Fig. 4 や Fig. 5 を見ればわかるように、点の集合の配置より PTAM が横材を認識できていることが分かる。これは、防撓材を通すための溝や、配管等のために開けられた穴により特徴点を認識しやすくなっているからだと思われる。そのため、穴や溝の存在しない縦材に対しては殆ど特徴点を認識しておらず、3DMap 上での判別は困難である。

また、模型以外の場所において認識されている特徴点が存在している。これはカメラを用いて広範囲を認識しているための不可避の事態ではあるが、対象以外で認識された特徴点は、形状認識という点においてはバグでしかない。従って、PTAM を形状認識に用いる際にはこれを取り除く必要があると考えられる。

特徴点の認識はカメラからの画像から角を認識するという処理を行っているため、対象の概形を捉えることは容易である。一方で、正確な寸法の計測には不向きである。そのため、形状認識として用いる際には、細かな部材点検等ではなく全体的な完成度を計測するのに適しているといえる。

実際の作業現場において形状認識をしようとする場合、カメラの移動は少なく、且つ全体像が正確に測定できる事が望ましい。そのため、どのくらいカメラを動かせば十分に形状認識ができるほど特徴点が取れるのかを検証し

た。結果、模型の 1 辺に沿って移動した場合には、十分に形状認識できるのは、最も手前側の横材のみであり、とても形状認識と呼べるものではなく、十分に模型を認識するためにはカメラを持って模型周辺を 1 周する必要があった。このモデルが 1/20 スケールであることを考慮すると、実空間での移動距離は膨大となり現実的ではない。対象以外の特徴点を検出する事と併せ、こちらも改善の余地ある。

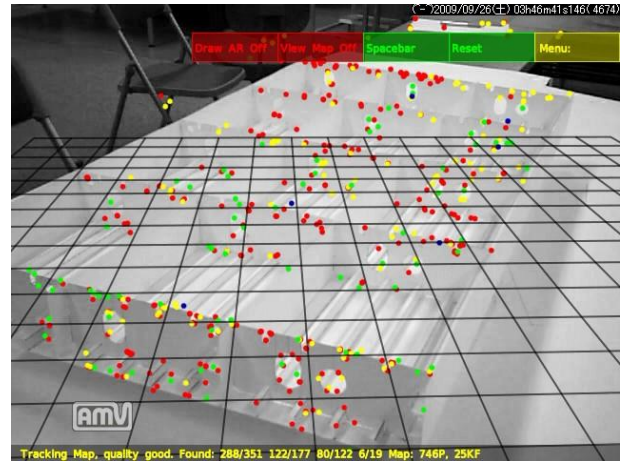


Fig. 4 Map points on model

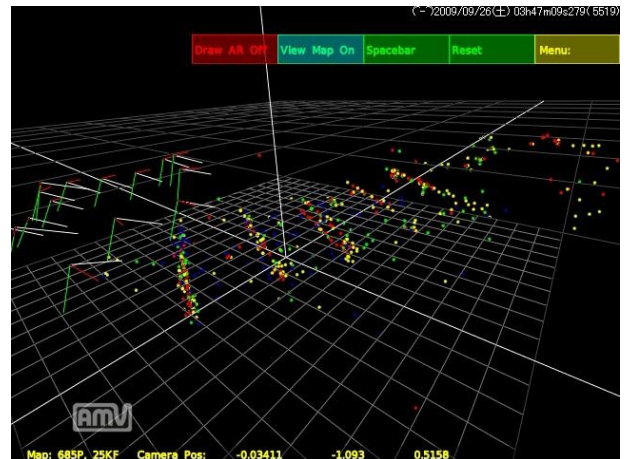


Fig. 5 Points of the 3D Map in PTAM

5. 結言

本論文では PTAM と呼ばれる手法の形状認識性能について評価を行った。ブロック模型の認識実験を通じて、意図しない特徴点の検出や凹凸の少ない平板からの特徴点未検出等、多くの問題点が判明した。しかし同時に、特徴点が認識できる物体に対しては、十分な性能で認識できる事を示した。

また、既に認識した特徴点の座標を外部出力できるようにプログラムを変更できているため、特徴点座標と 3D モデルデータを比較すれば、PTAM による形状認識の正確性を検証することができる。

前述の問題点を克服し正確な形状認識を行うことができれば、その結果を 3D モデルと形状比較することにより「建造進行状況の把握」や、「施工漏れの検出」等への使用可能性が考えられる。また、PTAM は AR 手法の一つであるため、認識環境中に実在しない物体を表示すること

ができる。この機能を用いれば、更に「今後の建造工程の視覚的認識」や「内装の検証」等にも応用が可能であると考えられる。

参 考 文 献

- 1) Georg Klein, David Murray : Parallel Tracking and Mapping for Small AR Workspaces, Proc. International Symposium on Mixed and Augmented Reality (ISMAR'7, Nara), 2007
- 2) H. Stewenius, C. Engels, D. Nister : Recent developments on direct relative orientation, ISPRS Journal of Photogrammetry and Remote Sensing, 60, 2006 June, pp. 284-294.
- 3) Edward Rosten and Tom Drummond : Machine learning for high-speed corner detection, European Conference on Computer Vision, 2006.
- 4) GLEW : <http://glew.sourceforge.net>
- 5) Lapack and BLAS :
<http://www.fi.muni.cz/~xsvobod2/misc/lapack/>
- 6) CMU 1394 camera driver :
<http://www.cs.cmu.edu/~iwan/1394/>
- 7) Pthreads : <http://sourceware.org/pthreads-win32/>
- 8) Libjpeg for windows :
<http://gnuwin32.sourceforge.net/packages/jpeg.htm>
- 9) Fltk : <http://www.fltk.org/>
- 10) EWCLIB :
http://www.geocities.jp/in_subaru/ewclib/index.html
- 11) PTAM for Webcam : 工学ナビの中の人の研究と周辺,
<http://kougaku.blog28.fc2.com/blog-entry-324.html>