6N-05

SLAM処理の並列分散化における最適な機能分割

松本拓也 † 大川猛 †† 大津金光 †† 横田隆史 ††
† 宇都宮大学工学部情報工学科 †† 宇都宮大学大学院工学研究科情報システム科学専攻

1 はじめに

警備，介護，極限環境作業ロボットにおいて自律移動ロボットの導入が期待されている．自律移動ロボットの開発ではバッテリーでの駆動を目的とした省電力化が必須である．またロボットが自律移動を行うために，地図の構築と自己位置推定を同時に行う技術として SLAM (Simultaneous Localization and Mapping) が研究されている．SLAM は測域センサや USB カメラといった外部センサと，ジャイロなどの内部センサから得られるデータを使い処理を行う．SLAM が行う処理 (以降 SLAM 処理) では，地図と自己位置の更新と修正を高精度で推定するために高い処理性能が要求される．

本研究では，ロボットの計算負荷の軽減と SLAM 処理で要求される高い処理性能を満たす方法として，SLAM 処理の一部をロボットの外部にあるサーバ上にオフロードし，サーバ側で並列処理することを提案する．本稿ではこの並列分散化を行うために SLAM 処理の機能分割の検討を行う．

2 Visual SLAM

Visual SLAM はカメラから得られる画像データのみを用いて自己位置推定と地図作成を行う方法である．SLAM にカメラを用いる利点 [1] としては，小型，正確，非侵襲的である．近年では安価かつ容易に手に入ることもあげられる．Visual SLAM は 2007 年に Davison らの MonoSLAM と呼ばれる単眼カメラのみを用いた SLAM によって初めて実装され [1]，2011 年には kinect に代表される RGB-D センサを用いた RGB-D SLAM が誕生し，注目を浴びている．

Visual SLAM の処理フローを図 1 に示す．Visual SLAM は画像データから特徴点を検出し特徴量を記述，その後特徴点マッチングと特徴追跡を行う．特徴追跡により算出したロボットの移動量は自己位置推定に用いる．こういったカメラ画像から移動量を推定する方法はビジュアルオドメトリと呼ばれる．またフレーム間のマッチングを行うことで地図の構築を行う．Visual SLAM はこれらの画像処理行い，さらに地図構築においては過去に得られた特徴量を用いてループ閉じ込み (一度訪れた場所を再認識すること) の検出の処理もあり，各段階で処理量が大きいという問題がある．

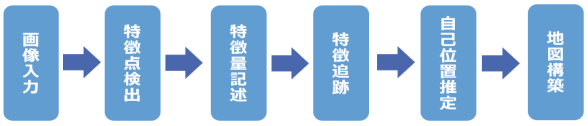

図 1: Visual SLAM の処理フロー

3 Visual SLAM の並列分散化

SLAM の中でも Visual SLAM は画像データを扱うためデータ量が大きく，また処理量も大きいためロボットの計算負荷が大きいという問題がある．そこで本研究ではそれらの問題を解決する方法として Visual SLAM の並列分散化を提案する．Visual SLAM の並列分散処理システムの概念図を図 2 に示す．画像のデータ量は特徴量記述により特徴量で表すことで削減可能である．このことを利用して SLAM 処理を前処理と後処理に分ける．例えば前者はロボット側が行う処理でセンシングや特徴量抽出，特徴量記述などデータ量が少なくなるまでの処理とする．後者は扱うデータ量と処理量の大きい自己位置推定や地図の構築などの処理をサーバ側の処理として分割する．このように SLAM 処理の並列分散化を行うためには，ロボット側とサーバ側との通信量や各処理の処理量を考慮して最適な機能分割を検討する必要がある．ここでいう「最適」とは，SLAM 処理で要求される高い処理性能を満たしながら，ロボットの計算負荷を最小限に軽減できるということである．

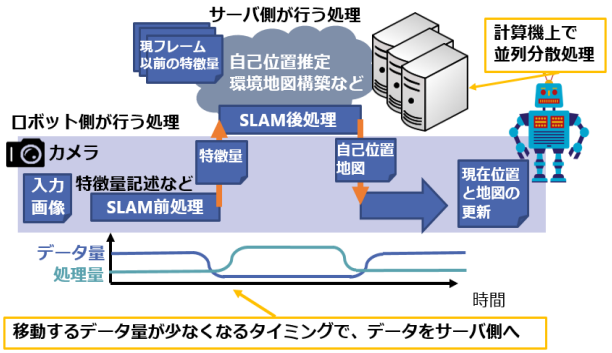

図 2: Visual SLAM の並列分散処理の概念

4 Visual SLAM の機能分割

Visual SLAM の最適な機能分割を検討するために，RTAB-Map (Real-Time Appearance-Based Mapping)[2] を対象とすることとした．RTAB-Map はグラフ構造を応用した RGB-D SLAM であり，セ

The optimum function partitioning in the parallel and distributed SLAM processing
†Takuya Matsumoto, ††Takeshi Ohkawa,
††Kanemitsu Ootsu, ††Takashi Yokota,
Department of Information Science, Faculty of Engineering, Utsunomiya University (†)
Department of Information Systems Science, Graduate School of Engineering, Utsunomiya University (††)

ンサから得られた RGB 画像と Depth 画像を元に，3D の地図をオンラインで構築する．また RTAB-Map は IROS2014 Microsoft Kinect Challenge で優勝した SLAM であり，ROS (Robot Operating System) のレポジトリにオープンソースで公開されている．

4.1 RTAB-Map の構成と性能分析

RTAB-Map を 1 台の PC 上で動作させ，その構成と性能を調べた．実験環境は，OS: Ubuntu14.04LTS, メモリ: 8GiB, CPU: Intel Core i7 4712MQ CPU @2.30GHz を，RGB-D センサには Microsoft 社の Kinect for Windows v1 を用いた．RTAB-Map のバージョンは 0.10.11 である．

RTAB-Map の主要なノード・トピックを選択して作成した処理フローを図 3 に示す (ノード・トピックの namespace は省略)．図中のノード名の下にある数値は，そのノードがメッセージを受け取ってから次のトピックへとメッセージを配信するまでの処理時間である．この処理時間は RTAB-Map を実環境で起動した後，100 秒間程センサをランダムに動かす動作を 3 回行った際の平均値 (100 秒間× 3 回) である．図 4 はトピックを流れるメッセージの使用帯域 (Byte/Sec) とトピックの配信頻度の平均値 (50 秒間× 3 回) を示している．トピックを流れるメッセージの使用帯域とトピックの配信頻度の調査には ROS の標準ツールである rostopic コマンドを使用した．

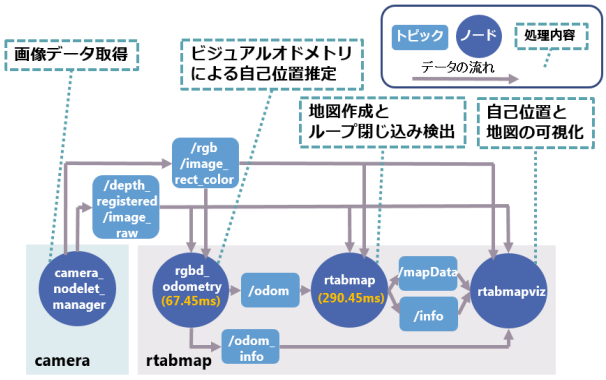

図 3: RTAB-Map の処理フローと処理時間

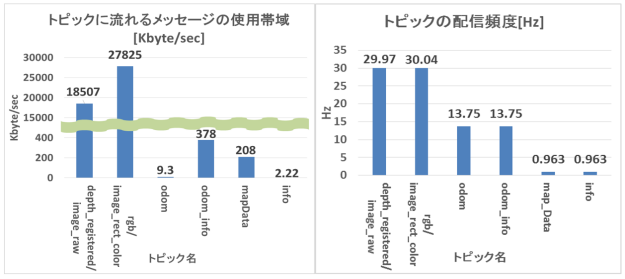

図 4: トピックを流れるメッセージの使用帯域 (Byte/Sec) とトピックの配信頻度 (Hz)

4.2 考察と機能分割案

性能分析より考案した機能分割案の処理フローを図 5 に示す．図 3 に示したように，rgbd_odometry ノードと rtabmap ノードが自己位置推定と地図作成を行っており，平均の処理時間はそれぞれ 67ms と 290ms かかる．この二つのノードをサーバ側に配置した場合，rgb/image_rect_color トピックと depth_registered/image_raw トピックを流れるメッセージの使用帯域がそれぞれ 28Mbyte/sec と 19Mbyte/sec あるため，ロボットとサーバ間の通信がボトルネックになる．この通信量を減らすためには，これらのトピックに流れる画像データを直接サーバ側に送るのではなく，記述した特徴量のみを送ればよい．また図 4 に示したように，rgbd_odometry ノードが配信する odom トピックと odom_info トピックの配信頻度は 13.75Hz なので，rgbd_odometry ノードが購読するトピックは 15Hz 程度あればよい．これらのことから特徴点検出と特徴量記述を 15fps 程度で行い，その直後の処理からをサーバ側との分割とすることが望ましい．

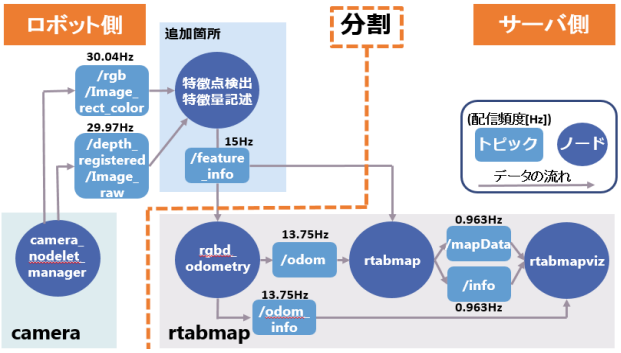

図 5: 機能分割案の処理フロー

5 おわりに

本稿では，Visual SLAM の並列分散化を提案した．また Visual SLAM である RTAB-Map の機能分割の検討を行った．今後は RTAB-Map を特徴量記述までの前処理と後処理とに分割し，処理性能を評価する予定である．

謝辞

本研究は一部総務省 SCOPE(No.0159-0112) および JSPS 科研費 24500055，15K00068 の助成による．

参考文献

[1] Davison,Andrew J., et al.: “MonoSLAM: Real-Time Single Camera SLAM.”, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol.29 No.6, pp.1052–1067, 2007.

[2] “RTAB-Map”, http://introlab.github.io/rtabmap/, (2016/1/5 アクセス).