Information Efficient 3D Visual SLAM in Unstructured Domains

Weizhen Zhou, Jaime Valls Miró and Gamini Dissanayake
ARC Centre of Excellence for Autonomous Systems
Mechatronics and Intelligent Systems Group
University of Technology Sydney
NSW2007, Australia
{*w.zhou, j.vallsmiro, g.dissanayake*}*@cas.edu.au*

Abstract

This paper presents a strategy for increasing the efficiency of simultaneous localisation and mapping (SLAM) in unknown and unstructured environments using a vision-based sensory package. Traditional feature-based SLAM, using either the Extended Kalman Filter (EKF) or its dual, the Extended Information Filter (EIF), leads to heavy computational costs while the environment expands and the number of features increases. In this paper we propose an algorithm to reduce computational cost for real-time systems by giving robots the 'intelligence' to select, out of the steadily collected data, the maximally informative observations to be used in the estimation process. We show that, although the actual evaluation of information gain for each frame introduces an additional computational cost, the overall efficiency is significantly increased by keeping the matrix compact. The noticeable advantage of this strategy is that the continuously gathered data is not heuristically segmented prior to be input to the filter. Quite the opposite, the scheme lends itself to be statistically optimal.

1. Motivation and Previous Work

One of the many important applications of mobile robots is to reach and explore terrains which are inaccessible or considered too dangerous for humans. Such environments are, for instance, frequently encountered in search and rescue scenarios where prior knowledge of the environment is unknown but required before any rescue operation can be deployed. A mobile robot equipped with an appropriate sensor package is probably the best aid for such scenario. The vehicle would be expected to autonomously navigate in six degrees of freedom (DoF) through a challenging rescue site, safely operating infield in order to generate a dense three-dimensional map of the environment understandable by human rescuers. The strategy hereby presented is motivated by this demanding application. While the formulation is indeed particularly suited to real-time systems in unstructured settings, it should be noted that is nevertheless not tied-up to any one domain.

In prior work [1], a novel sensor package along with a SLAM algorithm was proposed to meet the challenges described above. The sensor package consisted of a time-of-flight range camera (Swiss Ranger SR-2, low resolution, 160×240 pixels) and a higher resolution conventional camera (UniBrain Fire-i camera, 640×480 pixels). The two cameras were fixed relative to each other as illustrated in Figure 1. The conventional camera was used to capture scene texture and to extract salient visual features whilst the Swiss Ranger provided 3D range data of the corresponding scene. The combined observations made by these two cameras were used as the sole input in an EIF-based SLAM algorithm thereafter. It was assumed that no robot odometry information was available due to the unreliability of wheel encoder readings obtained from disaster scenes.

Fig. 1: Conventional pinhole camera aligned with Swiss Ranger

The existing technique employs a conventional EIF approach which recovers the robot and feature poses at the end of each 'acquire, update' cycle. The compulsory inversion of the information matrix evoked during each estimation cycle comes at a significant computational cost. Although the sensor package is capable of delivering data at a minimal frame rate of 10Hz, the increasing sampling delay between consecutive frames caused by extensive computation might yield inadequate data association and therefore unsuccessful frame registration. Such problem is particularly magnified in unstructured environment where robot's sights change rapidly and unpredictably along its undulating path.

Many efforts had been made in recent years to reduce the computational encumbrance generally faced by most SLAM

algorithms. In [3], Thrun *et al.* made the empirical observation that when feature-based SLAM posteriors were represented in information form, they were only dominated by a small number of links established between nearby features, therefore making many off-diagonal elements near zero when properly normalized. Later, Eustice *et al.* [5] suggested that such sparsification process tended to produce inconsistent estimates. Their proposal was an Exactly Sparse Delayed-State Filter (ESDSF) which maintains a sequence of delayed robot poses at locations where low-overlap images were captured. This approach allows an efficient representation without any sparse approximation error. In [6], the authors also used a Delayed State Extended Kalman Filter to fuse the data acquired with a 3D laser range finder. A segmentation algorithm was employed to separate the data stream - based on orientation restraints - into distinct point clouds, each referenced to a vehicle position. Both implementations significantly reduced the computational cost by eliminating features from the state vector to achieve practical performance. However, one noticeable common problem of these strategies is that loop closure can not be automatically detected. Separate loop closure methods were required in conjunction with their proposed techniques. Moreover, both methods require either human supervision over the acquired data or raw odometry measurements in order to minimise the number of critical robot poses that should be maintained. Yet none of these resources are normally readily available in settings such as the search and rescue scenario that motivates this work.

In this paper, we take an alternative approach to engage in the computational argument. Instead of gathering the minimal information and try to deal with it in the filter in the most efficient way from a purely mathematical standpoint, we seek an information-driven solution in which we allow the sensor package the freedom to collect data at its own rate, while giving the robot the 'intelligence' to choose those critical observations that should be incorporated in the estimation process. To that end, we extend the method first introduced in [1] and propose an improved filtering technique whereby given a desired estimation error bound, a buffer of overlapped frames are sampled but only the frame giving maximal information gain is added to the filter. By doing so, the filter incorporates a reduced number of robot poses optimally distributed along the robot trajectory based on uncertainty belief. Using this technique we can afford to maintain both the minimal set of robot poses with respect to the pre-set error bound, and their associated feature poses in the state vector in an efficient manner. Hence, making the case for producing more accurate maps when compared to strategies where only the robot poses are kept. Moreover, with the proper data association, the scheme lends itself to automatic detection of loop closure, be that by visual matching or Bayesian gating, or both.

The rest of this paper is structured as follows: Section 2 and 3 describe our data registration and the look-ahead and search backwards algorithm for buffering observations and generating base frames. Section 4 covers the mathematical formulations of the EIF SLAM algorithm. Section 5 displays the simulation results which compare a few key performance factors between the more traditional, computationally intensive approach, and our proposed methodology. Finally, we draw our conclusions in section 6 where improvement and future work is also discussed.

2. Feature Extraction and Frame Registration

Extraction of salient visual features and data association is carried out by the SIFT mechanism described in [4]. Firstly, SIFT features are detected in the 2D camera image and matched across those in the previous images. The corresponding pixels in the Swiss Ranger image are then derived, and the 3D position of these features is computed. Applying a least square 3D point set registration algorithm [2] and an outlier removal [7], the initial value of the new camera pose can be obtained with the estimated previous camera pose as prior. We refer the set of 3D points obtained at each frame as a scan. The details of this process are explained in [1].

3. Look-Ahead and Search Backwards Algorithm

Pseudo-code for the proposed algorithm is described in Algorithm 1. Assuming the robot begins at the origin $[0, 0, 0]$ of the global coordinate frame at time $t = 0$, the feature global poses can be established and be used as the first 'base' frame, F_{base}. For the following frames, matching features are found between F_{base} and each individual frame, unless the number of common features reaches a predefined minimum or the number of frames being examined exceeds the look-ahead buffer size. The minimal number of common features is restricted by the 3D registration algorithm [1] while the buffer size is an empirical number determined by the desired coarseness of the map. 3D registration is performed on each frame with respect to their matching F_{base} (all global coordinates). With the new observations made at each new robot pose, the information gain can be obtained without recovering the state vector which is the major computational expense in EIF SLAM. The camera pose at which the observations provide maximal information gain is added to the filter, and the corresponding frame is included as a new entry in the F_{base} database. Frames in the look-ahead buffer previous to the new F_{base} are dropped, and the new buffer starts from the consecutive frame after the newest entry in F_{base}. The same procedure is repeated until the end of the trajectory.

For the occasional circumstance when there are no matches between the frames in the look-ahead buffer and the frames in the F_{base} database, matching is attempted with the previously dropped frames. If a dropped frame provides sufficient matching features, it will be treated as a new frame and both will be updated in the filter. This mechanism, repeated for all the frames in the look-ahead buffer, ensures crucial information can be added back to the filter at any time shall the need arise.

The least desirable situation in search and rescue is when rescuers become rescuees due to the dangerous and unpredictable environment we intend to deal with. Although the

Algorithm 1 EIF Look-Ahead and Backward Search

```
while end of the image sequence not reached do
  if processing first frame, F_0 then
    Initialise iMatrix and iVector
    F_base = {F_0}
  else
    Set desired size of look forward buffer
    while End of look ahead buffer not reached do
      Do data association between F_i and F_base
      if More than 6 common features are matched then
        Get 3D registration of F_i
        if 3D registration successful then
          Init. new features, augment iMatrix and iVector
          Temporarily update filter wrt F_base and save
          determinant of new iMatrix
        end if
      end if
      F_i = F_{i+1}
    end while
    if At least one frame in look-ahead buffer then
      Find frame which updates iMatrix to have maximum
      determinant, F_j
      Update filter permanently with F_j
      F_base = {F_base, F_j}
    else
      Find match between frames in look-ahead buffer and
      last 5 frames not included in F_base
      if Match found then
        Update filter with two new frames
      else
        Kidnap recovery mode
      end if
    end if
  end if
end while
```

robot is not processing every frame it acquired during the rescue course, its comprehensive collection of the information about the environment became handy when kidnap situations occur as our robot not only has the knowledge of a set of known positions but also the information between these base positions. In real-life scenario, a robot may employ various of recovery strategies when kidnapped, being initialising a new map or trying to retrieve back to a known position. The proposed mechanism inevitably increases the chance for the robot to regain its location estimation based on past knowledge.

4. Extended Information Filter SLAM

Computational advantages of using an Extended Information Filter rather than an Extended Kalman Filter are now well known. This work employs an EIF that maintains all the features as well as the entire sequence of camera poses in the state vector. New camera poses are initialised with respect to the best matching frame at a known pose and measurement updates are additive in the information form. As we assumed the sensor package operates in full 6 DoF without a process model, the formulation of the filter becomes simpler and results in a naturally sparse information matrix: there is no motion prediction step to correlate the current state with its post and previous states.

For full 6 DoF SLAM, the state vector X contains a set of 3D features and a set of camera poses. The camera poses are represented as

$$(x_C, y_C, z_C, \alpha_C, \beta_C, \gamma_C) \tag{1}$$

in which α_C, β_C and γ_C represents the ZYX Euler angle rotation and the corresponding rotation matrix is referred to as $RPY(\alpha_C, \beta_C, \gamma_C)$. A 3D point feature in the state vector is represented by

$$(x_F, y_F, z_F) \tag{2}$$

expressed in the global coordinate frame.

Let i represent the information vector and I be the associated information matrix. The relationship between the estimated state vector $\hat{X}$, the corresponding covariance matrix P, the information vector i, and the information matrix I is

$$\hat{X} = I^{-1}i, P = I^{-1} \tag{3}$$

The first camera pose is chosen as the origin of the global coordinate system. At time $t = 0$, the state vector X contains only the initial camera pose $[0, 0, 0, 0, 0, 0]^T$, and the corresponding 6×6 diagonal information Matrix I is filled with large diagonal values representing the camera starting at a known position.

The observation model provides an estimation of the position of the new features by

$$\begin{pmatrix} \hat{x}_F \\ \hat{y}_F \\ \hat{z}_F \end{pmatrix} = \begin{pmatrix} \hat{x}_C \\ \hat{y}_C \\ \hat{z}_C \end{pmatrix} + \left(RPY(\hat{\alpha}_C, \hat{\beta}_C, \hat{\gamma}_C)^T \right)^{-1} \begin{pmatrix} x_L \\ y_L \\ z_L \end{pmatrix} \tag{4}$$

where x_L, y_L and z_L are the feature location expressed in the local reference frame.

In the update step, the information vector and information matrix update can be described by

$$\begin{aligned} I(k+1) &= I(k) + \nabla H_{k+1}^T Q_{k+1}^{-1} \nabla H_{k+1} \\ i(k+1) &= i(k) + \nabla H_{k+1}^T Q_{k+1}^{-1} [z(k+1) - \\ &\quad - H_{k+1}(\hat{X}(k)) + \nabla H_{k+1} \hat{X}(k)] \end{aligned} \tag{5}$$

where Q_{k+1} is the covariance matrix of the observation noise w_{k+1} and $z(k+1)$ is the observation vector.

The corresponding state vector estimation $\hat{X}(k+1)$ can be computed by solving a linear equation

$$I(k+1)\hat{X}(k+1) = i(k+1) \tag{6}$$

In error covariance form, the determinant of the $N \times N$ covariance matrix indicates the volume of the N-dimensional uncertainty polyhedron of the filter. The smaller the volume, the more confident the filter is about its estimation. As the information matrix has an inverse relationship with the covariance matrix, as described by (3), the maximally informative

TABLE 1: PERFORMANCE COMPARISON

Measurement	Consecutive	Look ahead 2	Look ahead 4
time(sec)	428.96	185.37	118.55
No frames	200	109	60
No features	96	95	90
Max Matrix Size	1488	939	630

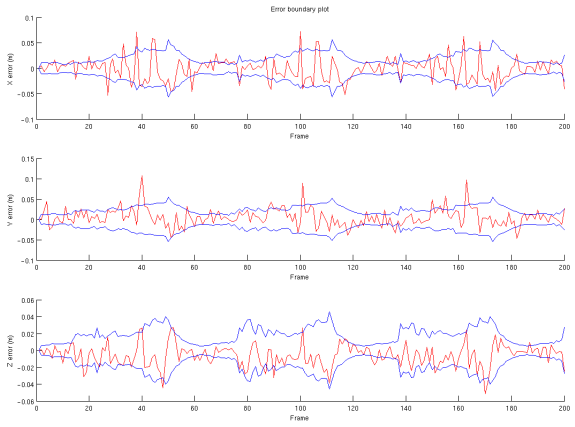


Fig. 2: Estimated covariance and estimation error in X, Y, Z coordinates, using 200 frames in the circular trajectory

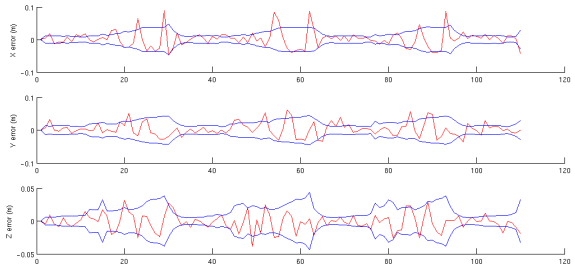


Fig. 3: Estimated covariance and estimation error in X, Y, Z coordinates, using 84 frames in the circular trajectory

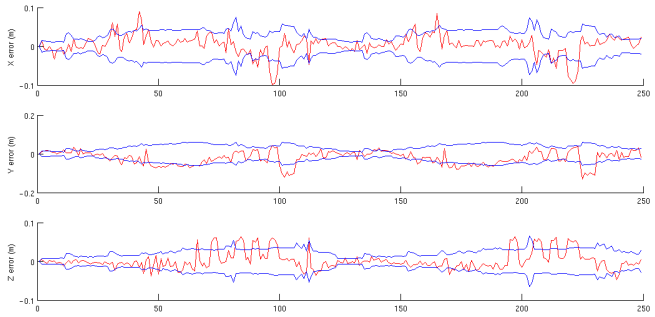


Fig. 4: Estimated covariance and estimation error in X, Y, Z coordinates, using 293 frames in the rectangular trajectory

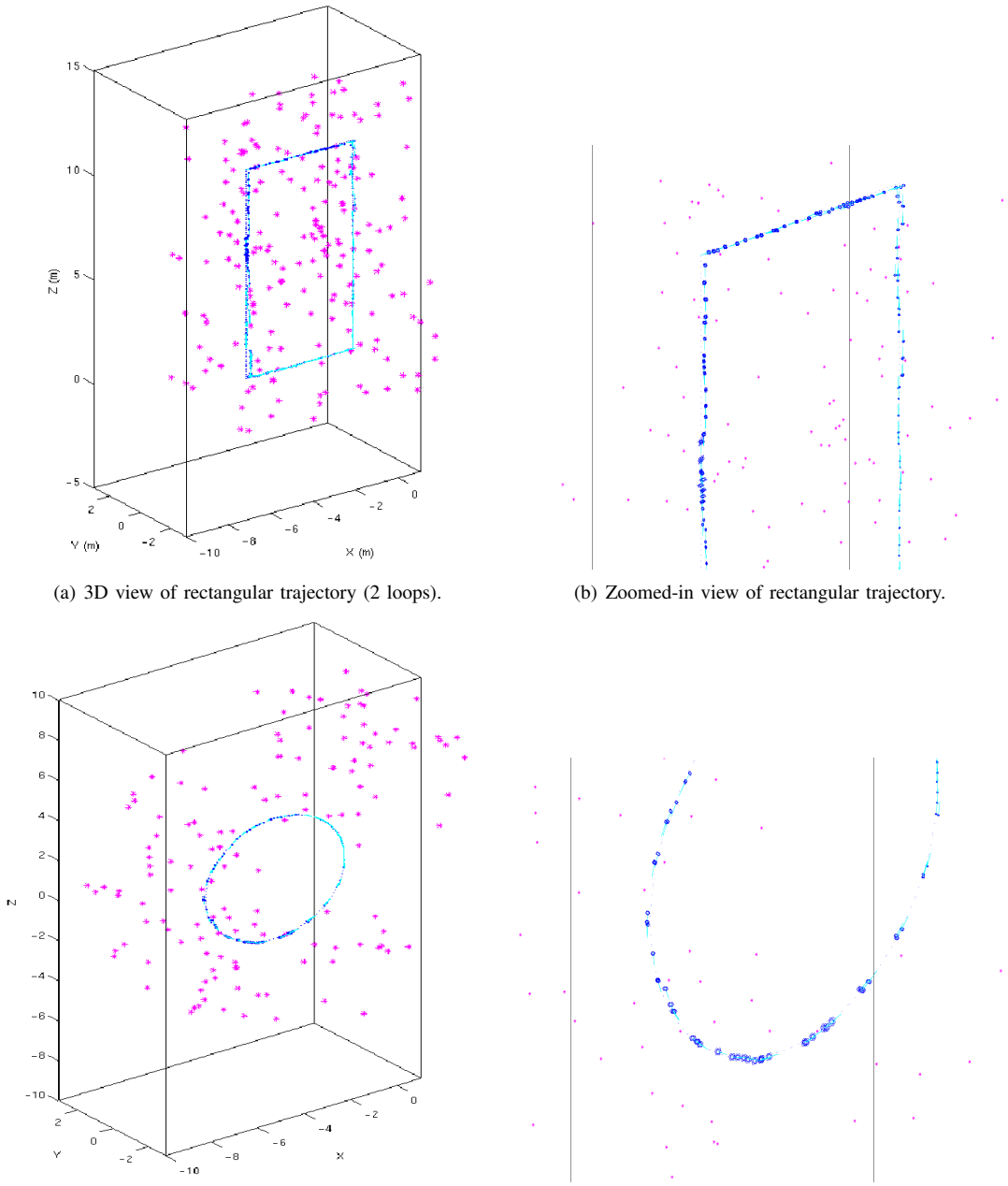
frame must update the information matrix to have the largest determinant. We use the natural logarithm of the determinant of the information matrix, denoted as $log(det(I(k+1)))$, as the measurable quantity of this information update. As described in Section 3, in a sequence of overlapped images containing common features, each image is evaluated with respect to the base frame database, F_{base}. Then, in order to proceed with the actual update of the filter, we choose the frame such that $log(det(I(k+1)))$ is maximised.

We also make use of an empirical threshold to gauge the update quality. When the maximum determinant is smaller than this threshold, meaning there is little information gain in updating the filter with the current sequence in the look-ahead buffer, we play safe and update all the frames in the sequence, hence maximising the information gain. This empirical threshold is defined based on the desired coarseness of the map.

5. SIMULATION RESULTS AND DISCUSSION

To validate the proposed strategy, we firstly test our algorithm in a simulation environment. In the following simple experimental setup, observations are made at 200 camera poses equally distributed along a circular trajectory of 3 loops and 3 meters radius. 500 features are randomly populated in the $10 \times 6 \times 10$ meters environment. Table 1 summarises the key performance between traditional EIF and the proposed technique. When all 200 frames are used to update the filter, the simulation completes in 428 seconds, 96 features are maintained. If we relax on information loss and the filter is allowed to look ahead in a buffer size of 3 frames, only 109 frames and 95 features are kept in the filter and the simulation completes in 185 seconds. When the filter is set to look ahead for longer (4 steps) only 60 frames are maintained with 90 features. Figure 2 shows the consistent estimation using conventional feature-based EIF while Figure 3 displays the similarly consistent result produced by the proposed algorithm with a buffer size of 3 frames. In another experimental setup, the robot is exercised along a rectangular trajectory of 10×5 meters, collecting data at 600 camera poses in 2 loops. With a buffer size of 3 frames, 248 frames are selected for filter update. The estimation error of this exercise is shown in Figure 4.

Figure 5(a) and 5(c) illustrate the 3-dimensional views of the estimated robot poses and their 95% uncertainty boundaries for the two experiments. Unlike most conventional fixed time step or fixed displacement approach, our proposed technique exhibits the ability to fuse the minimal information required based on the robot uncertainty belief, which in turn is dynamically influenced by the complexity of the robot trajectory, the observation quality and the estimation of the last known robot position. The results shown in zoomed-in Figure 5(b) are particularly relevant, as they clearly show how at sharp turns, such as the corners of the rectangular trajectory, the filter compensate for information losses caused by sudden changes in the scenes. As the robot pose uncertainty increases and information gets scarcer, a larger number of poses estimations are necessary to compensate for the reduced quality of the observations within the robot's field of view.



(a) 3D view of rectangular trajectory (2 loops).

(b) Zoomed-in view of rectangular trajectory.

(c) 3D view of circular trajectory (3 loops).

(d) Zoomed-in view of rectangular trajectory.

Fig. 5: Example trajectories depicting a rectangular and circular 3D motion. Dark (blue) ellipsoids show the robot's 95% uncertainty boundaries of the estimated trajectories. Light (aqua) lines show the robot current heading. Stars (pink) illustrate feature locations. In the zoomed-in view of the rectangular trejectory, it can be seen that more estimations were made after the sharp turn indicating the decrease of confidence level. Once the confidence was regained, the estimations became sparse along the trajectory.

6. Conclusion and Further Work

We have presented an approach for dynamically incorporating observations into the estimation process for 3D robot navigation in unstructured terrain. Results have shown that by gauging the information gain of each frame, we can automatically incorporate the most informative observations for the purpose of SLAM as well as obtaining a comprehensive collection of the information about the environment we intend to explore. The immediate next step is to test the proposed algorithm with the sensor package on a real-time robot platform. For real tasks in search and rescue scenarios, extracting the minimum information is only the first step towards efficiently understanding the spatial structure of the unknown environment. In future work, we intend to extend our study from *how to extract the optimal information* to *how to optimally extract such information*. As proposed in [8] and [9], active trajectory planning and exploration seem the natural companions to our proposed stragegy in order to

best comprehend the environment surrounding the robot, in minimum time and within bounded estimation errors.

ACKNOWLEDGEMENTS

This work is supported by the Australian Research Council (ARC) through its Centre of Excellence programme, and by the New South Wales State Government. The ARC Centre of Excellence for Autonomous Systems (CAS) is a partnership between the University of Technology Sydney, the University of Sydney and the University of New South Wales.

REFERENCES

[1] L. P. Ellekilde, S. Huang, J. Valls Miro, G. Dissanayake, "Dense 3D Map Construction for Indoor Search and Rescue", *Journal of Field Robotics*, vol. 24(1/2), pp. 71-89, 2007.

[2] K. S. Arun, T. S. Huang, S. D. Blostein, "Least Square Fitting of Two 3-D Point Sets", *IEEE Pattern Analysis and Machine Intelligence*, vol. 9(5), pp. 698-700, 1987.

[3] S. Thrun, Y. Liu, D. Koller, A. Y. Ng, z. Ghahramani, H. Durrant-Whyte, "Simultaneous Localization and Mapping with Sparse Extended Information Filters", *Int. Journal of Robotics Research*, vol. 23, pp. 693-716, 2004.

[4] D. G. Lowe, "Distinctive image features from scale-invariant keypoints" *Int. Journal of Computer Vision*, vol. 60(2), pp. 91-110, 2004.

[5] R. M. Eustice, H. Singh, J. J. Leonard, "Exactly Sparse Delayed-Sate Filters", in *Proceedings of the IEEE Int. Conference on Robotics and Automation*, pp. 2417-2424, 2005.

[6] D. M. Cole, P. M. Newman, "Using Laser Range Data for 3D SLAM in Outdoor Environments", in *Proceedings of the IEEE Int. Confenrence on Robotics and Automation*, pp. 1556-1563, 2006.

[7] M. Fischler, R. Bolles, "RANdom SAmpling Consensus: a paradigm for model fitting with application to image analysis and automated cartography", *Communications of the Association for Computing Machinery*, vol. 24, pp. 381-395, 1981.

[8] S. Huang, N. M. Kwok, G. Dissanayake, Q. P. Ha, G. Fang, "Multi-Step Look-Ahead Trajectory Planning in SLAM: Possibility and Necessity", in *Proceedings of the IEEE Int. Conference on Robotics and Automation*, pp. 1091-1096, 2005.

[9] R. Martinez-Cantin, N. de Freitas, A. Doucet, J. A. Castellanos, "Active Policy Learning for Robot Planning and Exploration under Uncertainty", in *Proceedings of Robotics: Science and Systems*, June 2007.