

Efficient Construction of Globally Consistent Ladar Maps using Pose Network Topology and Nonlinear Programming

Alonzo Kelly¹ and Ranjith Unnikrishnan²

¹ Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3890

alonzo@ri.cmu.edu

www.frc.ri.cmu.edu/~alonzo

² Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3890

ranjith@cs.cmu.edu

www.cs.cmu.edu/~ranjith

Abstract. Many instances of the mobile robot guidance mapping problem exhibit a topology that can be represented as a graph of nodes (observations) connected by edges (poses). We show that a cycle basis of this pose network can be used to generate the independent constraint equations in a natural constrained optimization formulation of the mapping problem. Explicit reasoning about the loop topology of the network can automatically generate such a cycle basis in linear time. Furthermore, in many practical cases, the pose network has sparse structure and the associated equations can then be solved time linear in the number of images. This approach can be used to construct globally consistent maps on very large scales in very limited computation. While the technique is applicable to mapbuilding in general, and even optimization in general, it is illustrated here for batch processing of 2D ladar scans into a mobile robot guidance map.

1 Introduction

Mapbuilding is an important problem in mobile robotics because triangulation from some form of map is often the only mechanism that enables stable accurate position estimates over either long time periods or long excursions. Regardless of the sensing modalities used, the environmental features used as landmarks, or the dead reckoning systems used to compute position between fixes to the map, the construction of a map which is sufficiently accurate and consistent remains a difficult problem.

This paper addresses the following problem. A sequence of ladar scans has been captured during a mapping pass through the environment. The imagery is tagged with estimated vehicle or sensor positions computed from all available dead reckoning indications, potentially including ladar scan matching (visual odometry). Based on these position estimates and an indication of which nonconsecutively captured scans overlap, a globally consistent map must be constructed.

1.1 Prior Work

This work is an adaptation of our earlier work on navigating from globally consistent appearance mosaics [1]. Building a map is closely related to the computer vision

problems of registering range scans to produce a geometric model or to registering imagery to produce mosaics.

The processing of large amounts of imagery into consistent representations was a core pursuit of analytic photogrammetry before there was computer vision or even computers [2]. The formulation of mosaicking (then known as block triangulation) as that of minimizing the total map residual using the normal equations was a basic technique. In the contemporary mosaicking literature, [3] is similar to the present work in its use of topological information to guide the optimization process. In the range image processing literature, we cite [4] and [5] for pioneering the iterative closest point algorithm, and [6] for using it to simultaneously minimize the total residual error of a number of overlapping range scans.

The idea of a network of reference frames connected by uncertain relative poses goes back in robotics at least to [7]. Many of the core concepts of consistency and constraint in uncertain geometric networks were introduced to robotics in [8]. These, in turn, have a partial basis in classical network theory [9].

The fact that tracking covariance leads to quadratic complexity in SLAM has been known since [10]. The fact that correlations must nonetheless be tracked was shown in [11]. These results clarify the value of our efforts to diagonalize covariance here. The fact that relative poses can do this was pointed out in [12].

One of the important aspects of this work is its ability to produce maps on a very large scale. In part, this is enabled by an inherently efficient formulation. Highly successful efforts to do the same for incremental approaches are also ongoing [13] [14]. In contrast to the hierarchical decomposition approach of the latter, our approach allows every image in the map to move in order to seek the constrained optimum; but otherwise, the use of paths through the network for formulating constraints on relative poses makes this work very similar.

Within the area of incremental mapping, global consistency of range scans was enforced in [15] by solving an overconstrained system of measurements to compute the required least-squares perturbation to absolute scan poses. This approach is extended in [16] to perform online mapping in cyclic environments. In contrast to these highly relevant works, our approach avoids the complexity of inverting large populated matrices in order to produce a method which is practical on a very large scale.

This document is organized as follows. Section 2 describes the constrained optimization framework underlying our proposed approach. Section 3 describes how an independent basis of constraint equations can be extracted automatically for a given map by using a graph representation. Results in large scale cyclic environments using this approach are presented in Sec. 4, followed by conclusions in Sec. 5.

2 Constrained Optimization

2.1 Constrained Optimization in a Relative Map

Consider a representation of the mapping problem in terms of relative poses. Our choice of relative poses for encoding spatial relationships assures that errors in

estimates of relative motion are decoupled. In the description that follows, the term *image* refers to the laser scan acquired at a given time instant, but may be used in a broader context to refer to any sensory observation of the world made relative to the observers viewpoint.

We denote the relative pose of image i with respect to image j by ρ_i^j where $\rho_i^j \in \text{SE}(2)$ in our application. As an example, an initial estimate for the relative pose $\hat{\rho}_{i+1}^i$ between temporally adjacent images numbered i and $i+1$ may be supplied by an odometer, but may also be independently supplied by (say) matching images acquired at those locations. In principle, if there are n images, up to n^2 relative poses can be defined. In practice, only those relative poses for which an important overlap exists need to be considered.

We also denote the homogenous transform relating the coordinate frame associated with image j with respect to that at image i by T_j^i . Note that this transform is purely a function of the associated relative pose. We denote the uncertainty of an estimate of ρ_i^j by P_{ij} . The overcomplete vector of initial estimates of relative poses thus has a diagonal initial state covariance matrix, which is denoted as P_0 .

A change in the topology of the map occurs when the path of a robot closes on itself, or the when robot records an observation at a location previously encountered along its trajectory. Such an event introduces constraints in an overcomplete representation of the map as the true values of relative poses describing motion between those locations are now constrained to be consistent. In this formulation, internal consistency is enforced by hard constraints on the relative poses, termed *loop equations*, which take the form:

$$T_{i+1}^i T_{i+2}^{i+1} \dots T_{i+j}^{i+j-1} T_i^{i+j} = \mathbf{I} \quad (1)$$

Such equations require the total transform around a closed loop to be the identity. Likewise, consistency with external reference points may be imposed with *path equations* of the form:

$$T_{i+1}^i T_{i+2}^{i+1} \dots T_{j-1}^{j-2} T_j^j = T_j^i|_{\text{external}} \quad (2)$$

A loop is simply a closed path in these expressions and the path is therefore the more general notion.

Let the state vector \mathbf{x} denote the vector of all relative poses for which an important overlap exists. The square total of the residual $\mathbf{z}(\mathbf{x})$ of matching points or features in all image pairs serves as a useful measure of performance for optimization purposes. Note that the residual between observations of a feature position made in images i and j is only a function of the relative pose ρ_i^j and the relative coordinates of the feature observed in each image.

We may then specify the problem as:

$$\begin{aligned} \text{Minimize: } & f(\mathbf{x}) = \frac{1}{2} \mathbf{z}(\mathbf{x})^T \mathbf{z}(\mathbf{x}) + \frac{\mu}{2} (\mathbf{x} - \hat{\mathbf{x}})^T P_0^{-1} (\mathbf{x} - \hat{\mathbf{x}}) \\ \text{subject to constraints: } & \mathbf{c}(\mathbf{x}) = \mathbf{b} \end{aligned} \quad (3)$$

The second term in the objective function penalizes the Mahanolobis distance between the true value of relative pose and its initial estimate, and hence implicitly

$$\begin{bmatrix} H^T H + \mu P_0^{-1} & C^T \\ C & 0 \end{bmatrix}$$

Fig. 1. Closing a Single Loop With Nonlinear Programming. Shaded cells denote non-zero elements. The diagonal part of the matrix is all that needs to be inverted.

assumes a Gaussian generative model for relative pose errors. It is well known that over the course of robot motion, the uncertainty in pose estimates using only odometry deviates significantly from a Gaussian distribution. However, over the very small distance traversed between two successively acquired images, a Gaussian remains a valid approximation of odometry error. A similar argument can be applied for the error in a relative pose estimate after a scan matching operation. Hence we claim that the error metric is defensible under this problem formulation. The free scalar μ simply influences the relative weighting between the two quantities in the objective function.

Such problems are, of course, solved using Lagrange multipliers. We adjoin the constraints to the objective function to form the Lagrangian:

$$l(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^\top [\mathbf{c}(\mathbf{x}) - \mathbf{b}] \quad (4)$$

and require of a constrained minimum that:

$$\begin{aligned}\frac{\partial l}{\partial \mathbf{x}} &= \frac{\partial f}{\partial \mathbf{x}} + \boldsymbol{\lambda}^\top \frac{\partial \mathbf{c}}{\partial \mathbf{x}} = \mathbf{0} \\ \frac{\partial l}{\partial \boldsymbol{\lambda}} &= \mathbf{c}(\mathbf{x}) - \mathbf{b} = \mathbf{0}\end{aligned}\tag{5}$$

To generate the constrained Newton’s method, we linearize these equations about the current estimate \hat{x} where they are not satisfied, transpose where convenient, and require satisfaction upon perturbation. The ultimate result of this manipulation is:

$$\begin{bmatrix} H^T H + \mu P_0^{-1} & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \lambda \end{bmatrix} = - \begin{bmatrix} H^T \mathbf{z} \\ \mathbf{c}(\hat{\mathbf{x}}) - \mathbf{b} \end{bmatrix} \quad (6)$$

where the Jacobians H and C are evaluated at the current estimate as:

$$H = \frac{\partial z}{\partial x}|_{x=\hat{x}} \qquad C = \frac{\partial c}{\partial x}|_{x=\hat{x}}$$

These equations can be iterated until convergence from a initial estimate in order to produce the constrained local minimum. Note that this system of equations is similar to the basic form of the Variable State Dimension Filter (VSDF) [17]. The matrix $H^T H + P_0^{-1}$ is block diagonal as shown in Fig. 1, and the solution to the system can be obtained directly by first solving for λ in:

$$C \left(H^T H + \mu P_0^{-1} \right)^{-1} C^T \boldsymbol{\lambda} = [\mathbf{c}(\hat{\mathbf{x}}) - \mathbf{b}] - C \left(H^T H + \mu P_0^{-1} \right)^{-1} H^T \mathbf{z}(\hat{\mathbf{x}}) \quad (7)$$

and substituting back to solve for Δx in:

$$(H^T H + \mu P_0^{-1}) \Delta \mathbf{x} = -(H^T \mathbf{z}(\hat{\mathbf{x}}) + C^T \boldsymbol{\lambda}) \quad (8)$$

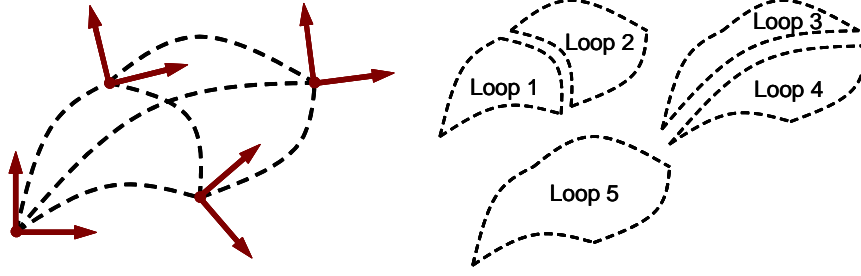


Fig. 2. Network Cycle Basis: There are 4 nodes in this graph and 6 edges. Notice that there is no node at the center of the figure so the edge intersection there is of no consequence. There are five distinct loops but only three of them are independent.

2.2 Complexity of the Optimized Relative Map Formulation

The first equation is solved for the Lagrange multipliers and the second is then solved for the state perturbation. As illustrated in Fig. 1, the matrix to be inverted in this case is block diagonal. Let n be the number of relative poses, l be the number of independent constraints and m be the total number of features considered in all image pairs. With careful implementation, the complexity of these computations is $O(m + nl^2 + l^3)$ for the first equation and $O(m + nl)$ for the second. Note that the highest power of n that appears is unity.

It is the decision to use a representation of potentially inconsistent relative poses which leads to a nearly diagonal system of equations. The alternative of using a minimal independent set of poses to represent the map would require, essentially, that the constraints be substituted into the objective. Doing so leads to a fully populated coefficient matrix. When the constraints on the poses are represented in a second set of equations as hard constraints, the resulting structure can be solved much more efficiently. The next section describes the process of generating this set of constraint equations that are required to enforce consistency in the map.

3 Pose Network Analysis

In the general case, the generation of independent loop equations is itself a difficult problem. The main concern is that the loop equations must be independent to preserve the rank of the coefficient matrix CC^T . For extracting this set of equations automatically, we use a graphical representation of the map as described below.

We define the pose network as a graph with $G = (\mathcal{V}, \mathcal{E})$. Each node $\vartheta \in \mathcal{V}$ corresponds to a time instant when an image was observed. For every directed edge ϵ from node i to j , $\epsilon \in \mathcal{E}$ only if an independent relative pose estimate exists between images at i and j . This may occur in one of two ways:

- a) if i and j are adjacent in time, and an estimate of relative motion is available as ρ_i^j through (say) odometry, or

- b) if the images corresponding to nodes i and j contain common features that make direct estimation of ρ_i^j possible.

An independent cycle basis of this graph then corresponds to a set of independent constraint equations, with each cycle in the basis corresponding to a loop equation in the set.

Figure 2 illustrates an example where there are 4 nodes (shown with their associated reference frames) and 6 edges. Although there are nine distinct loops, four of them can be eliminated because there is no node in the center of the figure. The remaining five loops include those formed by each set of three nodes, and the one loop composed of four nodes. Of these five loops, only three are independent.

Generating the loop equations requires an algorithm for computing the cycle basis of the network. While the literature on this aspect of network theory devotes significant effort to finding the best basis for a given purpose [18], our results do not depend on the exact basis chosen.

3.1 Extracting a Cycle Basis

In general, the problem of finding a cycle basis is related to the problem of finding a spanning tree of the network. Each different possible spanning tree corresponds to a different cycle basis. In a spanning tree, each node (except an arbitrarily chosen root node) is assigned one and only one parent. Once the tree is constructed, two important statements become true:

- There is now a unique acyclic path between any two nodes in the tree.
- Any edge in the network, but not in the tree, closes an independent loop which is a member of the cycle basis encoded by the tree.

Figure 3 illustrates this on a simple tree. The choice of root node and the order of traversal determines the form of the tree. Notice that if there are N nodes, then the spanning tree contains $N - 1$ edges. If there are E edges in the original network, then there must be:

$$L = E - (N - 1) = E - N + 1 \quad (9)$$

independent loops. In many practical scenarios, the number of images $N \gg L$.

The spanning tree can be generated by a breadth first traversal of the pose network. For this purpose, edges are considered bidirectional. Starting from the root, each node and edge is marked as it is traversed. When any node is encountered for the second time:

- The edge traversed is marked as a loop closing edge and not included in the spanning tree.
- The node is not elaborated into its children.

When there are no nodes left in the traversal queue, the tree is complete. This algorithm is clearly of complexity E because each edge is visited only once.

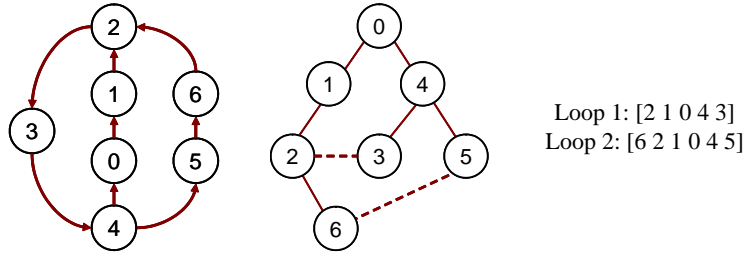


Fig. 3. Spanning Tree and Cycle Basis: The network of 7 nodes has two independent loops. The spanning tree generated when node 0 is chosen as the root is illustrated. The edges between nodes (5,6) and (2,3) are the identified loop closing edges.

The generation of loop constraint equations requires the generation of an explicit sequence of edges which comprise each loop in the cycle basis. A sequence of loop edges is comprised of the path in the spanning tree and the edge outside the tree which closes the loop. It is important to track whether the edges in the loop are traversed in the same sense or the opposite sense of the relative pose to which they correspond.

The shortest path in the spanning tree between the nodes associated with a loop-closing edge is easy to find, and is implicitly coded in the parent links of the spanning tree. In total, this step of finding all loop equations from their corresponding edge-sequences has a complexity bounded from above by the length of the largest loop times the number of loops. Therefore, in no case can it exceed LE operations.

4 Results

The constrained optimization mapping system has been recently implemented and is undergoing field trials. Presently, images are captured roughly ten times a second. With reasonable odometry, there is little difference in the quality of maps produced using the full constrained optimization approach versus simply enforcing the constraints without optimization. The latter amounts to a right pseudoinverse applied to all loops simultaneously. Consecutive scans are registered using the iterative closest point (ICP) algorithm before the mapping process commences. Computing times required to register consecutive scans in this manner are not included in the times provided because these calculations can be performed on-line while the images are being captured. The processor used is a 1.5 GHz Pentium with 1 GB of RAM. The two example trials provided below have been chosen to exemplify how very large maps can be rendered globally consistent in comparatively little computation.

4.1 Large Cyclic Facility

One trial involved mapping several floors of Pittsburgh's ice hockey stadium. A map of one floor is presented below both before and after rendering it consistent.

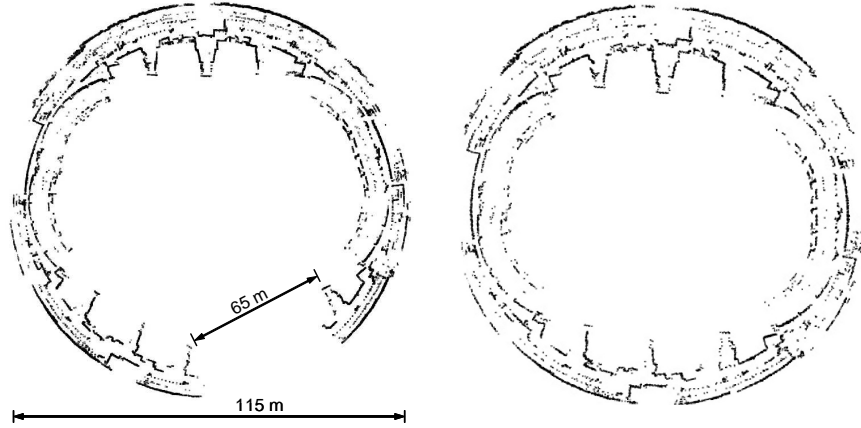


Fig. 4. Ladar Map of Pittsburgh's Mellon Arena. 3875 ladar scans have been rendered globally consistent in 0.14 seconds of computation. Left: original odometry-based map. Right: final map

Clearly, a slight curvature error to the right seems evident in the odometry. In this case, four iterations were required to reduce the original 65 meter error at loop closure to under 1 cm. Overall runtime was 0.14 seconds to just enforce constraints and 0.65 seconds to also compute the optimum map.

4.2 Large Multi-Cyclic Facility

The odometry was very well calibrated in this second test. Here, the environment is a typical grocery store. Such environments are ideal for demonstrating the benefits of our approach. There are a large but finite number of loops and the number of images required to map the facility is enough to overwhelm many alternative approaches. The vehicle trajectory was a left to right scan of the facility where each set of shelves was encircled once.

The left figure testifies to the excellent odometry calibration in effect for this test. After 11 complete turns and roughly 700 meters of travel, the heading error remains under 20° . This error does, however, account for the gradual skew in the aisles as the vehicle moved from left to right in the figure. Misregistration at the ends of the aisles is evident when they are seen a second time after looping around each set of shelves. Various fuzzy clouds of points in the aisles are either displays or occasional moving people.

The quality of the odometry led to an average misregistration on the order of 3 meters at the point of closure of each of the 23 loops used. Four iterations were required to enforce all constraints to under a cm in 12.22 seconds of computation. For problems on this scale, the extra effort to compute an optimal map is insignificant because the constraint equations require most of the processing.

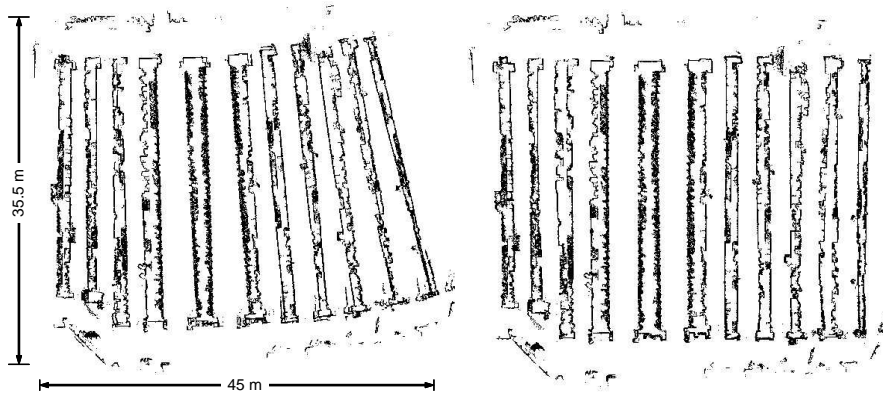


Fig. 5. Ladar Map of 12 Aisles of a Grocery Store. 10,380 ladar scans have been rendered globally consistent in 12.22 seconds of computation. Left: original odometry-based map. Right: final map

5 Conclusions

This paper has shown the promise of explicit topological reasoning to solve the problem of globally consistent mapping on the large scale. When the pose network describing observations is sparse, as it often is, and the problem size is large, the computational advantages in terms of both memory and processing can be both substantial and enabling with respect to comparable batch-processing techniques.

The technique is applied here to ladar scans and in our earlier work to image mosaicking [1], but it applies in principle to any form of imagery that can be registered whether it is 2D or 3D. Indeed, the fundamental result applies to all constrained optimization problems, and it is likely to have been exploited in other fields for some time.

At an abstract level, sparse cyclic systems are systems which are only slightly overconstrained. For such systems, the process of extracting an explicit expression of the hard constraints encoding consistency seems to have great practical value. Sparse optimization problems then become reducible to computationally easy problems in graph theory followed by easy ones in optimization.

While we have demonstrated the use of the technique in a batch processing mode, it is amenable to incremental map building. Indeed, its linear complexity implies that optimal consistent maps of large scale facilities could be constructed on the fly as long as the cycle basis can be maintained and a robust automatic mechanism for detecting loop closure can be developed. The latter has been shown by several other researchers (e.g. [19]), and the former can be accomplished by simply re-running the present algorithm as new loops in the pose network are closed.

To the degree that limited sensor range and environmental occlusion makes all problems sparse on the large scale, we have begun to support the case that globally consistent mapping can be accomplished in a fully automatic fashion on very large

scales without requiring any additional assumptions in order to reduce computational complexity.

References

1. R. Unnikrishnan and A. Kelly, "A Constrained Optimization Approach to Globally Consistent Mapping", *Proc. International Conference on Robotics and Systems (IROS) 2002*.
2. M. Thompson, R. Eller, W. Radlinski, and J. Speert, editors, "Manual of Photogrammetry", American Society of Photogrammetry, 3rd Edition, 1966.
3. H.S. Sawhney, S. Hsu and R. Kumar, "Robust video mosaicing through topology inference and local-to-global alignment", *Proc. European Conf. on Computer Vision (ECCV)*, Freiburg, Germany, vol.2, pp.103-119, June 1998.
4. Z. Zhang: "Iterative Point Matching for Registration of Free-Form Curves and Surfaces", *International Journal of Computer Vision*, 1994.
5. P. Besl and N. McKay: "A Method of Registration of 3-D Shapes", *IEEE Trans. PAMI* 12, 1992.
6. Bergevin, Soucy, Gagnon, and Laurendeau: "Towards a General Multi-View Registration Technique", *IEEE Trans. PAMI* 18, 1996.
7. R. Smith, M. Self, and P. Cheeseman, "Estimating Uncertain Spatial Relationships in Robotics", in *Autonomous Robot Vehicles*, Springer Verlag, New York, 1990. I. Cox, and G. Wilfong, editors.
8. H.F. Durrant-Whyte, "Integration, Coordination, and Control of Multi-Sensor Robot Systems", chapter 2, Kluwer, 1988.
9. Chartrand, "Graphs as Mathematical Models", Pringle 1977.
10. P. Motalier, R. Chatila, "Stochastic multisensory data fusion for mobile robot location and environmental modelling", *5th Symposium on Robotics Research*, Tokyo, 1989.
11. J.A. Castellanos, J.D. Tardos, and G. Schmidt, "Building a Global Map of the Environment of a Mobile Robot: The Importance of Correlations", *Proc. IEEE Conference on Robotics and Automation*, pages 1053-1059, 1997.
12. P. Newman, "On the Structure and Solution of the Simultaneous Localization and Map Building Problem", *Ph.D. thesis submitted in March 1999, Australian Centre for Field Robotics, Univ. of Sydney*.
13. J. Guivant and E. Nebot, "Optimization of the Simultaneous Localization and Map Building Algorithm for Real Time Implementation", *IEEE Trans. on Robotics and Automation*, June 2001, 17(3), pp. 242-257
14. J. Leonard, and H. Feder. "A Computationally Efficient Method for Large-Scale Concurrent Mapping and Localization", in *Proc 9th International Symposium on Robotics Research*
15. F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping", *Autonomous Robots*, 4:pp.333-349, 1997.
16. J.S. Gutmann and K. Konolige, "Incremental Mapping of Large Cyclic Environments", *Proc. Computational Intelligence in Robots and Automation (CIRA) 1999*, pp. 318-325.
17. P.F. McLauchlan, "The Variable State Dimension Filter applied to Surface-based Structure from Motion", *CVSSP Technical Report VSSP-TR-4/99*
18. N. Deo, G.M. Prabhu and M.S. Krishnamoorthy, "Algorithms for generating fundamental cycles in a graph", *ACM Trans. Math. Software*, 8 (1982), pp. 26-42.
19. D. Hähnel, S. Thrun, B. Wegbreit and W. Burgard, "Towards Lazy Data Association in SLAM", *Proc. of the 11th International Symposium of Robotics Research (ISRR)*, 2003.