

Exploring The Effect of Meta-Structural Information on the Global Consistency of SLAM

Mina Henein, Montiel Abello, Viorela Ila and Robert Mahony¹

Abstract—Accurate online estimation of the environment structure simultaneously with the robot pose is a key capability for autonomous robotic vehicles. Classical simultaneous localization and mapping (SLAM) algorithms make no assumptions about the configuration of the points in the environment, however, real world scenes have significant structure (ground planes, buildings, walls, ceilings, etc..) that can be exploited. In this paper, we introduce meta-structural information associated with geometric primitives into the estimation problem and analyze their effect on the global structural consistency of the resulting map. Although we only consider the effect of adding planar and orthogonality information for the estimation of 3D points in a Manhattan-like world, this framework can be extended to any type of geometric, kinematic, dynamic or even semantic information. We evaluate our approach on a city-like simulated environment. We highlight the advantages of the proposed solution over SLAM formulation considering no prior knowledge about the configuration of 3D points in the environment.

I. INTRODUCTION

In order to perform autonomous tasks such as surveying and inspection, search and rescue within man-made environments, autonomous driving in urban environments, and reconnaissance, a robot must be able to simultaneously build an accurate representation of the surroundings and localize itself. Simultaneous localization and mapping (SLAM) and structure from motion (SfM) algorithms are used for this purpose. SLAM algorithms have to be run on-board the robotic platform and provide a globally and structurally consistent solution at every step. Building such consistent maps is an essential capability for robots autonomously navigating the environment and performing tasks. In the SLAM community, there is no universally accepted notion of global structural consistency [18]. Algorithms that are highly accurate (minimizing residual errors) can still admit large structural inconsistencies that contravene relatively simple global geometric structure such as vertical walls, flat floors, etc.. In particular, the returned map may drift very far from the true global structure of the environment. We refer to such a situation as the algorithm displaying poor global structural consistency. Classical SfM and SLAM algorithms usually make no assumptions on the structure of the scene being analyzed [24]. In practice, however, these algorithms are applied to scenes that contain significant structure that could be exploited.

The contribution of this paper is to explore the effect of adding meta-structural information into the estimation problem, and evaluate the structural consistency of the results. In particular, we exploit planar information applied to 3D points that are known *a-priori* to lie on planar surfaces. We also exploit plane orthogonality between detected planes. The plane parameters are incorporated into the estimation problem as additional random variables that are dependent on the environment points, and/or other planes but are not directly observed by the robot. We show that by utilizing prior knowledge of the environment, more accurate and globally consistent solutions can be obtained.

The remainder of this paper is structured as follows, in the following section we discuss the related work. In section III we describe the proposed approach for adding prior spatial information. In section IV we introduce the experimental setup, followed by the actual experimental results and evaluations in section V. We summarize and offer concluding remarks in section VI.

II. RELATED WORK

The earliest works on SLAM were based on the extended Kalman filter (EKF) approach [3] [16] [2]. However, it has been shown that filtering is inconsistent when applied to the inherently non-linear SLAM problem [9]. One intuitive way of formulating SLAM is to use a graph representation with nodes corresponding to the random variables (robot poses and/or landmarks in the environment) and edges representing functions of those variables (typically measurement functions). Lu and Milios [17] first proposed the graph-based formulation of the SLAM problem in 1997. Once such a graph is constructed, the goal is to find a configuration of the nodes that is maximally consistent with the measurements [7]. Approaching SLAM as a nonlinear optimization on graphs has been shown to offer very efficient solutions to moderate scale SLAM applications [8] [14]. Factor graphs, such as the one in Fig.1, are graphical models that have been used for representing the SLAM problem [5] [12] [11]. This is due to the fact that in factor graphs the functions are made explicit and such a bipartite graph is directly connected to the solutions of the optimization problem [11].

One of the earliest works that consider adding geometrical constraints to improve the quality of the reconstruction is presented by Szeliski and Torr [24] where they hallucinate additional point matches based on image homographies which are either given directly or computed between a collection of *a-priori* known co-planar points. Zucchelli et al. [30] showed how linear constraints among feature points

¹ Mina Henein, Montiel Abello, Viorela Ila and Robert Mahony are part of the “Australian Centre of Excellence for Robotic Vision”, Research School of Engineering, The Australian National University, Canberra ACT 2601, Australia mina.henein@anu.edu.au, u5162235@anu.edu.au, viorela.ila@anu.edu.au, robert.mahony@anu.edu.au

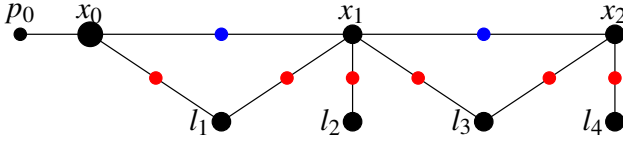


Fig. 1: Factor graph representation for SLAM problem. Variable nodes include poses x_i and point features l_j . Factor nodes corresponding to the odometric measurements o_i in blue, the point measurements z_k in red and the prior p_0 on the first pose in black (best seen in color).

(e.g. co-linearity and co-planarity) can be incorporated in the minimization process to improve the structure from motion estimates from optical flow using least squares minimization of the differential epipolar constraints. Several plane parameterizations are used in the literature. Lee et al. [15] use a graph formulation in combination with a spherical parameterization of the planes to extract planes and use them to correct odometry between consecutive frames. Another approach presented by Triebel and Burgard [28], extracts constraints from 3D range scans and uses them for pose estimation in a graph-based SLAM framework. Another common approach to represent a plane by four parameters is to use its normal and distance to the origin. Trevor et al. [27] uses this over-parameterized representation for a smoothing solution. The over-parameterized formulation is also used by [25] for real-time mapping, combining both mapping of points and planes.

While some work has been done on using planar constraints for mapping, less work has been addressing a least-squares SLAM estimation with planar constraints in the literature. Most recently, De la Puente and Rodriguez-Losada [4] presented an approach to perform landmark SLAM optimization, along with different level structure detection (points, segments, lines and circles) in an Expectation-Maximization algorithm, where only the last robot pose and the features belong to the graph. Kaess [10] also shows how to formulate SLAM that directly estimates infinite planes instead of the 3D landmarks in the environment. It uses homogeneous plane parameterization with a corresponding minimal representation for the optimization which is suitable for use with NLS incremental solvers. Our work on the other hand, estimates for both, structure points and planes, this allows for 3D reconstruction of more realistic environments where planar and nonplanar structures coexist.

III. PROBLEM FORMULATION

A. Graph-based SLAM problem formulation

In visual SLAM (e.g. where localization and mapping are done using visual sensors) or SFM problems, the goal is to estimate the 3D structure of the environment and the camera poses that maximally satisfy a set of measurement constraints [20].

It has been shown in the SLAM literature [6] [21] that Gaussian noise models lead to computationally efficient solutions. Recent advances in the field propose to formulate

the SLAM problem as a nonlinear least squares (NLS) optimization. This is done over a set of variables; the camera/robot poses $\mathbf{x} = \{x_0 \dots x_{nx}\}$, with $x_i \in se(3)$ and $i \in 0 \dots nx$; and the 3D points $\mathbf{l} = \{l_1 \dots l_{nl}\}$ where $l_j \in \mathbb{R}^3$ and $j \in 1 \dots nl$. Together these variables constitute the state $\theta = [\mathbf{x}, \mathbf{l}]$ of a size $n = nx + nl$ of a classical landmark SLAM problem.

Two types of measurements, the odometry obtained by robot's proprioceptive sensors and the observations of the landmarks in the environment obtained by processing the images from an on-board camera are typically integrated into a visual SLAM. Let $f_i(x_{i-1}, x_i)$ be the odometry model with Σ_{v_i} , odometry noise covariance matrix:

$$o_i = f_i(x_{i-1}, x_i) + v_i, \quad \text{with } v_i \sim \mathcal{N}(0, \Sigma_{v_i}) \quad (1)$$

where $\mathbf{o} = \{o_1 \dots o_{mi}\}$, $o_i \in se(3)$ is the sequence of odometric measurements. Similarly, let $h_k(x_{i_k}, l_{j_k})$ be the 3D point measurement model with Σ_{w_k} , the measurement noise covariance matrix :

$$z_k = h_k(x_{i_k}, l_{j_k}) + w_k, \quad \text{with } w_k \sim \mathcal{N}(0, \Sigma_{w_k}) \quad (2)$$

where $\mathbf{z} = \{z_1 \dots z_{mk}\}$, $z_k \in \mathbb{R}^3$ is the set of all 3D point measurements.

The solution of the problem is obtained by minimizing the sum of squared nonlinear residuals:

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \frac{1}{2} \left\{ \sum_{i=1}^{mi} \|f_i(x_{i-1}, x_i) - o_i\|_{\Sigma_{v_i}}^2 + \sum_{k=1}^{mk} \|h_k(x_{i_k}, l_{j_k}) - z_k\|_{\Sigma_{w_k}}^2 \right\} \quad (3)$$

Iterative non-linear optimization methods such as Gauss-Newton or Levenberg-Marquardt can be used to find a solution that minimize (3). At each i^{th} iteration, the cost function in (3) is linearized and the solution is found by solving a linear least-squares (LS) problem in δ [5]:

$$\delta^* = \underset{\delta}{\operatorname{argmin}} \frac{1}{2} \|A^T A \delta - A^T \mathbf{b}\|^2, \quad (4)$$

where A gathers the derivatives of the residuals in (3) with respect to variables in θ weighted by the block-diagonal matrix that gathers all the square rooted, inverse covariances of all the observations; and \mathbf{b} is the residual evaluated at the current linearization point. The new linearization point is obtained by $\theta^{i+1} = \theta^i + \delta^*$. This formulation is often used in the SLAM literature [5] [12] [14] [20]. The graph formulation of the SLAM is highly intuitive and has the advantage of being able to incorporate several types of observations (odometry, GPS, IMU, sonar, laser scan registration, feature points, feature lines, etc., even reprojection errors in the case of SFM). Another advantage of this formulation is that it allows for efficient implementations of batch [14] [1] [13] and incremental [12][11] [19] NLS solvers.

B. Adding planar constraints to a graph-based SLAM

In this paper, we explore the effect of integrating additional information about the landmarks in the environment into the visual SLAM estimation problem. While this paper is

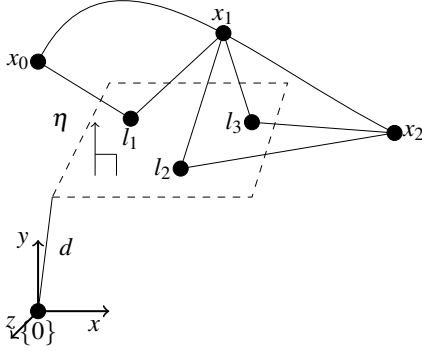


Fig. 2: SLAM with planar information representation. Camera poses are represented by x_i , landmarks by l_j and plane parameters by η and d .

only concerned with constraining some points to planes and applying orthogonality or parallelism constraints between planes, the formulation is general and can be extended to any type of structural constraints present in the environment as long as there is an *advanced front-end* that can provide that information and a function that can model it. This is a common situation in “built environments” where planar surfaces are ubiquitous. The front-end can either incrementally process the estimated map of points and detect planes and angles between planes or use the raw image input to detect and track planes.

A plane is characterized by the normal to the plane surface $\eta_s \in S^2$ with $S^2 = \{(\eta_x, \eta_y, \eta_z)^\top \mid \|\eta\| = 1\}$ the unit sphere in space, and the distance to the planar surface $d_s \in \mathbb{R}$, with $s = 1 \dots ns$ and ns is the total number of planes detected. For each point l_j that lies on a plane defined by $p_s = [\eta_s, d_s]^\top$ we have:

$$0 = g(l_j, p_s) + q_{s_j} \quad \text{with} \quad g(l_j, p_s) = d_s - l_j^\top \eta_s, \quad (5)$$

where q_{s_j} is $q_s \sim \mathcal{N}(0, \Sigma_{q_{s_j}})$ normally distributed zero-mean Gaussian noise. Here q_{s_j} models the true distance of the point from the plane, that is the formulation does not require the underlying plane model to hold exactly, such as for example in the case where points in a corridor may lie on pin-boards on the wall and vary slightly in depth, or points on the buildings lay on the window frames, etc. We chose to parameterize a plane using η_s and d_s , although it is an over-parameterization, because in this way, it is easiest to represent the distance of a point to the plane in the way stated above, and also implies a mathematically simple jacobian computation.

Furthermore, given two planes with η_{s1} and η_{s2} , an angle constraint can be defined as:

$$\cos(a_t) = \eta_{s1}^\top \eta_{s2} + r_t, \quad (6)$$

where r_t is the normally distributed zero-mean angle noise with covariance matrices Σ_{r_t} . Here r_t models the confidence

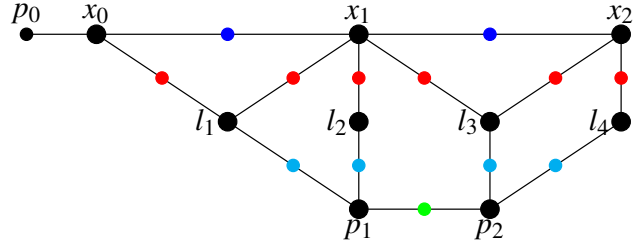


Fig. 3: Factor graph representation for SLAM problem with added planar and angular information. New variables s_p are added to incorporate the planes as well as the new factors, g_{s_j} in cyan and a_t in green (best seen in color).

that is assigned to plane orthogonality or parallelism constraint a_t .

C. The cost function

The factor graph representing a SLAM problem which integrates two planar constraints and one angle constraint is shown in Fig. 3. This factor graph representation helps us to formulate the nonlinear least square problem that minimizes all the residuals defined by the integration of the meta-structural information into the estimation problem:

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \frac{1}{2} \left\{ \sum_{i=1}^{mi} \|f_i(x_{i-1}, x_i) - o_i\|_{\Sigma_{v_i}}^2 + \sum_{k=1}^{mk} \|h_k(x_{i_k}, l_{j_k}) - z_k\|_{\Sigma_{w_k}}^2 + \sum_{s,j}^{ms} \|d_s - l_j^\top \eta_s\|_{\Sigma_{q_s}}^2 + \sum_{t=1}^{mt} \|\cos(a_t) - \eta_{s1}^\top \eta_{s2}\|_{\Sigma_{r_t}}^2 \right\} \quad (7)$$

where mi , mk , ms and mt are the number of odometric measurements, point measurement, plane observations and angle observations.

D. Particularities of the Jacobian matrix

Taking the derivative of $g(l_j, \eta_s, d_s) = d_s - l_j^\top \eta_s$ with respect to $\eta_s \in S^2$ involves projecting the differential of $g(l_j, \eta_s, d_s)$ onto the tangent space of S^2 using the projector $(I - \eta_s \eta_s^\top)$:

$$\nabla_{\eta_s} g(l_j, \eta_s, d_s) = (I - \eta_s \eta_s^\top) \frac{\partial g(l_j, \eta_s, d_s)}{\partial \eta}. \quad (8)$$

Observe that representing a plane by four parameters is an over-parameterization, which causes the system matrix to be singular with zero eigenvalues associated with the term $(I - \eta_s \eta_s^\top)$ and any attempt to invert it within the NLS solver will fail.

There are two ways to overcome this problem. The parameterization of the plane can be characterized by η_s now in \mathbb{R}^3 and $d_s \in \mathbb{R}$, in which case the optimization should constrain $\eta_s^\top \eta_s - 1 = 0$ to ensure the solution converges to a unit norm of the normal vector η_s . This can be modeled as a unary factor in the factor graph in Fig. 3. This approach not only introduces extra unnecessary factors in the graph

for each connected plane but it is also very sensitive to the confidence associated to this prior.

A second and more elegant approach is to keep the $\eta_s \in S^2$ parameterization and to solve a constrained optimization problem. This can be done by calculating the perpendicular kernel [22] to the matrix $A^\top A$ and solve the equation $(K^\perp)^\top A^\top A K^\perp \delta' = (K^\perp)^\top A^\top b$ and then reconstruct the $\delta = K^\perp \delta'$. Appendix VI explains how this is applied to solve the NLS in (3).

IV. EVALUATION SETUP

The framework is evaluated on Manhattan-like worlds, where objects and structures typically exhibit a high degree of organization in the form of orthogonal and parallel planes [23]. In order to evaluate the structural consistency of the proposed methodology, we generated several datasets with ground truth (GT). This is done by simulating a robot moving in a Manhattan-like world and observing 3D points in the environment using a sensor with a limited field of view. Odometric measurements are also available. We also emulate an advanced front-end that is able to fit planes to the observed 3D points, and planes are assumed to be either orthogonal or parallel.

A. Simulated dataset

To clearly observe the effects of adding structural information, we simulate a robot moving forward on a long trajectory without revisiting previous locations. As a baseline, the system is solved with no additional structural information (NoI) integrated into the estimation. This is shown in subfigure a) in both Fig. 4 and 5. We then consider two different configurations of planes to model the buildings on the left and right sides of the road. In the first case, all buildings on each side are represented as a single plane. In the second case, four planes are generated on each side to more accurately and realistically model separate buildings on a street. These two cases are repeated when planar information (PI) is added to the system, subfigures b) and d) and when both planar and angular information (PAI) is added, subfigures c) and e) in both Fig. 4 and Fig. 5. For each of these cases, the same odometry and point measurements are used to provide a valid comparison. This comparison is repeated with ten sets of randomly generated measurements. Experiments were conducted with combinations of several noise levels for odometry, point measurement, plane constraints and angle between planes. The noise levels are shown in the captions of the corresponding figures.

B. Synthetic city dataset

The second dataset is an existing synthetic Blender-generated city by Zhang *et al.* [29], from which we generated a dataset which consists of RGB-D images acquired by a robot traveling in this synthetic city. Ground truth depth can be associated to every pixel in the image. It is worth mentioning that this dataset provides a good model of a realistic environment, with the only difference of having perfect depth estimates and ground-truth camera

poses. To realistically model a real case scenario, we deliberately assign high noise levels to odometry and points measurements ($\Sigma_v = \text{diag}[0.4m, 0.4m, 0.4m, 6^\circ, 6^\circ, 6^\circ]^2$, $\Sigma_w = \text{diag}[0.4m, 0.4m, 0.4m]^2$). A feature matching and tracking algorithm was run to generate a set of 3D point measurements. On top of that we implemented a plane extraction algorithm that uses M-estimator SAmple Consensus (MSAC), a variant of the RANdom SAmple Consensus (RANSAC). Only the planes that satisfy the Manhattan world assumption are integrated into the estimation. Two experiments were run with this dataset, one where the robot stops before closing a loop (62 poses and 550 3D points) and a second one with a loop closure (135 camera poses and 1500 3D points). Experiments were conducted with combinations of several noise levels for odometry, point measurement, plane constraints and angle between planes. The noise levels are shown in the captions of the corresponding figures.

C. Implementation details

We implement an estimation framework which is able to integrate not only simple point measurements but also meta-structural information. The object oriented design is thought to accommodate future extensions to different types of objects and even to integrate dynamics of the scene. The estimation is implemented as a solution to an NLS problem as presented in section III. The solver communicates with an advanced front-end which provides the structural constraints. Batch and incremental processing is possible. The estimation is solved using Levenberg-Marquardt method.

V. EXPERIMENTAL RESULTS

This section evaluates the proposed technique on the datasets described in section IV. We are focused on analyzing the accuracy and structural consistency of the proposed estimation solution and compare it to the classical SLAM formulation which does not integrate any additional information about the 3D points in the environment. In this sequence of experiments, we consider points constrained to planes, and orthogonality and parallelism information between planes, that is using $a_t = 90\text{deg}$ or $a_t = 0\text{deg}$. All the experiments are run on an Intel Core i7-4790 CPU machine.

A. Accuracy tests

The accuracy is evaluated by comparing the absolute trajectory translational error (ATE), the absolute trajectory rotational error (ARE), the absolute structure error (ASE), the relative trajectory translational error (RTE), the relative trajectory rotational error (RRE), and the relative structure error (RSE). The absolute error is used to evaluate the structural consistency of the estimation.

Numbers in Table. I & II show better accuracy results in case of added structural information. More analysis follows in subsections V-B & V-C.

Simulated Street Dataset					
Error	NoI*	PI*	PAI*	SPI*	SPAI*
ATE	1.42 m	0.74 m	0.74 m	0.89 m	0.83 m
ARE	0.33 °	0.12 °	0.12 °	0.20 °	0.18 °
ASE	2.28 m	0.93 m	0.92 m	1.21 m	1.11 m
RTE	0.05 m	0.05 m	0.05 m	0.05 m	0.05 m
RRE	0.21 °	0.16 °	0.16 °	0.16 °	0.16 °
RSE	0.70 m	0.66 m	0.66 m	0.67 m	0.67 m

TABLE I: Average error values for 106 experiments of the simulated dataset. * NoI: without added information, PI: planar information added, PAI: planar & angular information added, SPI and SPAI are the same as PI and PAI respectively in the case where each side of the street is modeled as several planes.

Synthetic City Dataset		
Error	NoI*	PAI*
ATE	1.28 m	0.65 m
ARE	5.79°	2.44°
ASE	1.29 m	0.63 m
RTE	0.31 m	0.30 m
RRE	1.35°	1.21°
RSE	0.68 m	0.50 m

TABLE II: Error values for the synthetic city dataset without loop closure. * NoI: without added information, PAI: planar & angular information added.

B. Analysis of the simulated street dataset

Simulations were run for 16 different combinations of point measurement and plane flatness noise levels, with 10 different randomly generated measurement datasets, giving a total of 160 trials. Due to space limitations we only show results for 2 representative trials: full results figures and the associated code can be found on our Entity SLAM website <https://github.com/MinaHenein/Entity-SLAM>.

For the trials shown in both Fig. 4 and Fig. 5, the point measurement noise level σ_w is set to be relatively high in order to better observe the performance advantages of our algorithm.

From the tests we observe that planar information helps to preserve the structural consistency of the estimated map. This is most obviously seen in the reduction in drift between a) where no structural information is used, to the b) where walls are modeled as single planes, in both Fig. 4 and Fig. 5 and also reflected in table I. There are some cases where this planar information results in increased drift. The inherent randomness of the measurement noise model can result in the estimation drifting back towards the ground-truth. A close examination of Fig. 5 a) shows that the estimated trajectory initially drifts towards the negative in the x axis, but this is corrected by a later drift. Fig. 5 b) shows that the planar information tends to preserve the straightness of the structure,

and thus this reversal of direction is not allowed. This situation is unlikely to occur in a real environment, where a city is more realistically modeled by separate planes. When considering the case where walls/buildings are modeled as several planes, the system is more susceptible to drift because of the increased freedom allowed by segmentation of the planes. In fact, if the number of planes used to represent one building is sufficiently high, the added planar information act as the no information case; as if every point had the freedom to lie on a single plane. The addition of angular information helps to reduce this effect, as this additional information reshapes the problem as a single plane. This can be seen by comparing d) and e) in both Fig. 4 and Fig. 5. Table I shows 47% error reduction in ATE, 59% reduction in ASE and 63% reduction in ARE when considering 1 plane on each side (PI & PAI) and 37% error reduction in ATE, 46% reduction in ASE and 38% reduction in ARE for planar information when considering segmented planes (SPI) and 41% error reduction in ATE, 51% reduction in ASE and 44% reduction in ARE for planar and angular information when considering segmented planes (SPAI). When including angular information, care should be taken when choosing the strictness of the constraints, as the system can become sensitive to large measurement noise resulting in failure of the estimation. The corrections provided by the angle information must not exceed freedom in plane fitting allowed by the spatial distribution of the points. This is reinforced by our observation that experiments where points are generated loosely on the planes as in Fig. 5 the estimation is less susceptible to failure, as opposed to very close to planes as in Fig. 4. Improving the robustness of this formulation is an ongoing direction of research.

It is worth mentioning that on average there is a 16.3% increase in computational time when incrementally running *PI* tests with respect to the *NoI* tests and another 2.5% increase when adding angular constraints. Note the current MATLAB implementation used to perform the tests was not optimized for performance.

C. Analysis of the synthetic city dataset

Although run for two different setups of experiments, and four different noise levels each, in this paper we only show results from one experiment and a single noise level, full results figures can be found at the web site mentioned above.

It is known that loop closures help to preserve structural consistency when the same locations are revisited and the same environment landmarks are re-observed and used to correct the drift. Therefore in the city dataset, adding planar and angular information appear to have minimal effect after the loop is closed.

Improvements are however shown in the case where the robot traversed the city but no large loops have been closed yet. It is here where the estimate drifts by a significant amount without added meta-structural information. Using prior knowledge of the environment in terms of planar and angular information improves the overall estimation accuracy and preserves consistency. Fig. 6 shows the effect of adding

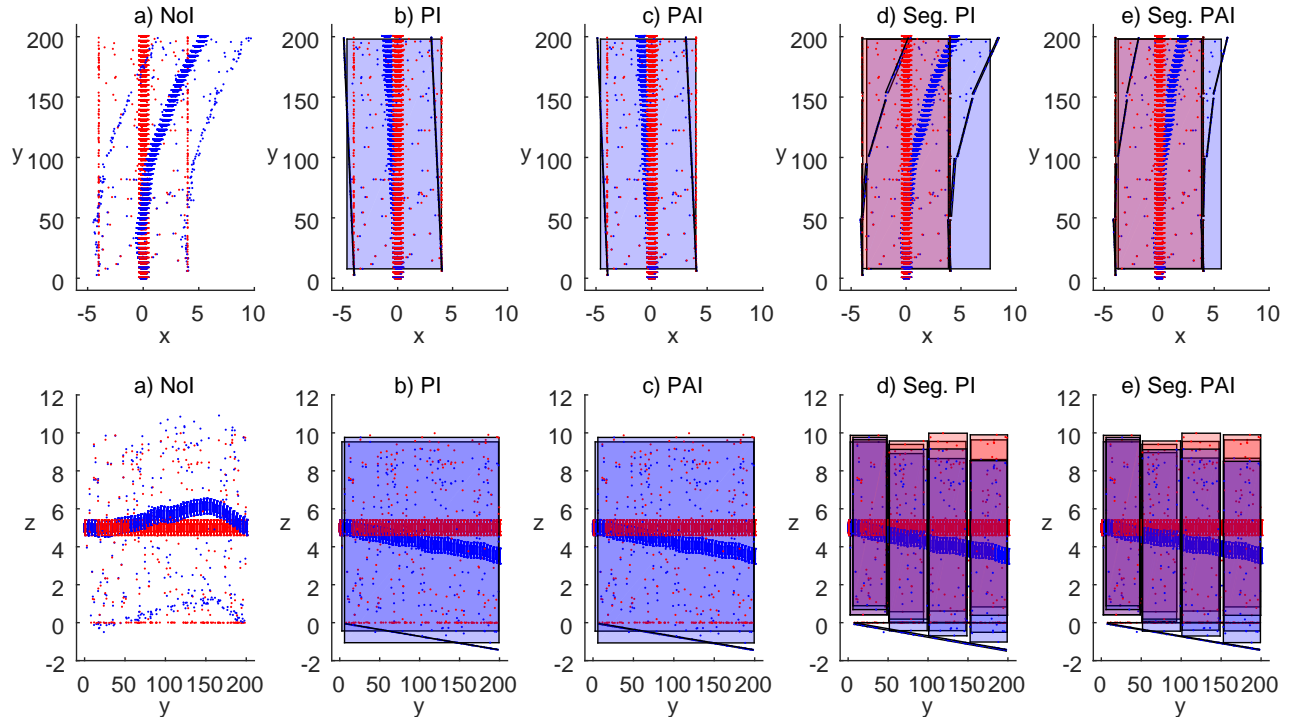


Fig. 4: Comparison of SLAM with a) no added structural information, with added planar information when modeling walls as b) one plane and d) four separate planes, and with both planar and angular information when modeling walls as c) one plane and e) four separate planes. The Second row shows a different view of the same scene. Ground-truth is shown in red and the final estimation is shown in blue. The figure is not displayed to scale. Noise levels used are: $\Sigma_v = [0.02m, 0.02m, 0.02m, 2^\circ, 2^\circ, 2^\circ]^2$, $\Sigma_w = [0.2m, 0.2m, 0.2m]^2$, $\Sigma_g = [0.01m]^2$ and $\Sigma_r = [0.05^\circ]^2$. Figure best seen in color.

planar and angular information on the global consistency of the estimation, and Table II shows a reduction of 50% in absolute errors.

VI. CONCLUSION AND FUTURE WORK

In this paper, we have shown the effect of additional meta-structural information on the accuracy and global structural consistency of SLAM. We used the factor graph formulation of a SLAM problem to incorporate prior knowledge of the environment into the estimation. Results show improvements in the estimation of structure, 3D points pertaining to a certain surface show better convergence results versus the same points with no added information.

The proposed algorithm demonstrates better convergence and global structural consistency results in most cases and still can be improved in many aspects. A possible improvement can be made by having plane detection integrated into the estimation process or ran as a prediction step [10]. It is also worth mentioning that angular information should be added with care, as solving for angular constraints has shown to be very sensitive to initial conditions and can sometimes cause the problem to get stuck in a local minima. This has happened in 33% of the 160 total experiments run for the simulated street dataset (Table I does not include those cases).

While presented in isolation here, the planar and orthogonality information can be combined with other geometric or semantic information as well as dynamics of the objects in the scene to handle a larger group of problems. Different information can be added to the system such as distance information for geometric shapes of known dimensions or velocity information for moving rigid objects, this can greatly improve the incremental segmentation SLAM problem [26] and opens wide doors to SLAM with objects and dynamic SLAM estimation problems.

APPENDIX

We start with the linear system described by the equation:

$$A^T A \delta = A^T b, \quad (9)$$

with optimal solution δ^* . We define K , a matrix whose columns span the null space of $A^T A$, and K^\perp an orthogonal matrix to K and whose columns span the domain of $A^T A$ by construction. The size of A is $M \times N$, where $N = \dim(x_i) \cdot nx + \dim(l_j) \cdot nl + \dim(\eta_s) \cdot ns + \dim(d_s) \cdot ns$, $M = \dim(o_i) \cdot mi + \dim(z_k) \cdot mk + \dim(p_{s_j}) \cdot ms + \dim(a_t) \cdot mt$, therefore $K \in \mathbb{R}^{N \times ns}$ and $K^\perp \in \mathbb{R}^{N \times (N - ns)}$. Since the solution δ^* lies in the domain space of $A^T A$, we can write $\delta^* = K^\perp \delta^{*'}$, where $\delta^{*'} \in \mathbb{R}^{N - np}$. Thus one has:

$$A^T A \delta^* = A^T A K^\perp \delta^{*'} = A^T b. \quad (10)$$

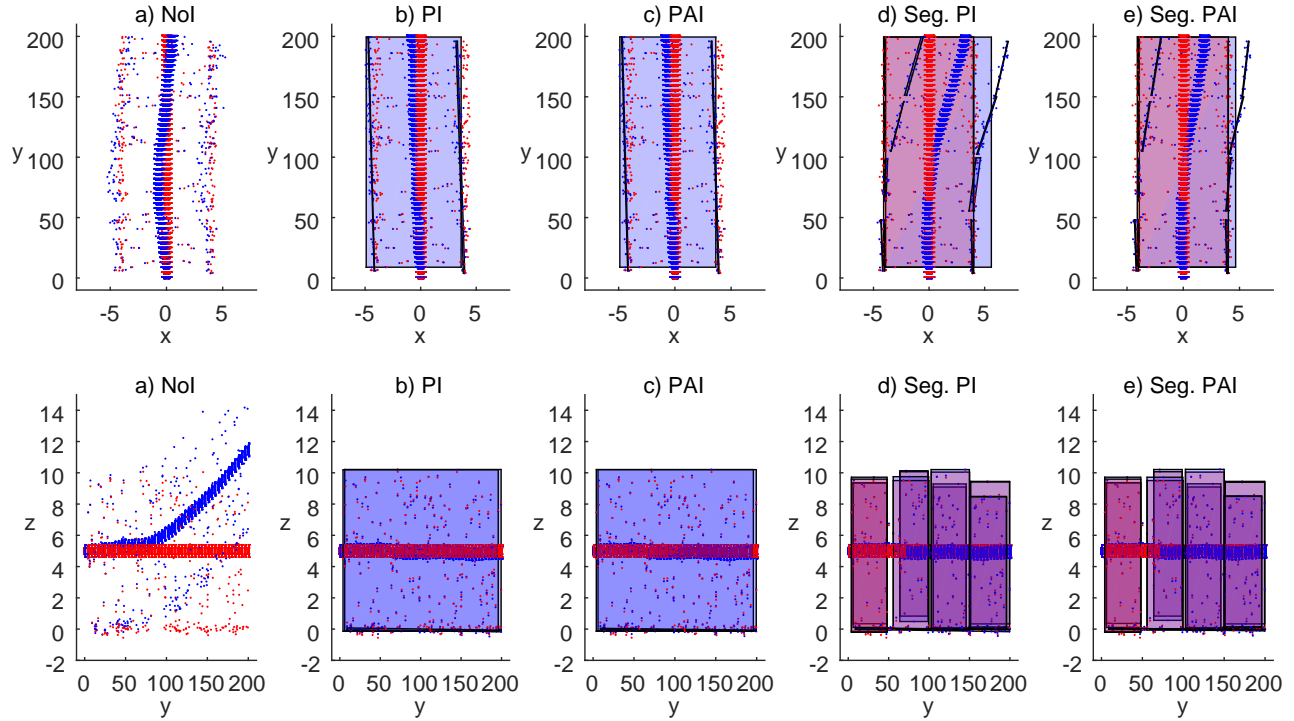


Fig. 5: Comparison of SLAM with a) no added structural information, with added planar information when modeling walls as b) one plane and d) four separate planes, and with both planar and angular information when modeling walls as c) one plane and e) four separate planes. The Second row shows a different view of the same scene. Ground-truth is shown in red and the final estimation is shown in blue. The figure is not displayed to scale. Noise levels used are: $\Sigma_v = \text{diag}[0.02m, 0.02m, 0.02m, 2^\circ, 2^\circ, 2^\circ]^2$, $\Sigma_w = \text{diag}[0.2m, 0.2m, 0.2m]^2$, $\Sigma_g = [0.2m]^2$ and $\Sigma_r = [0.05^\circ]^2$. Figure best seen in color.

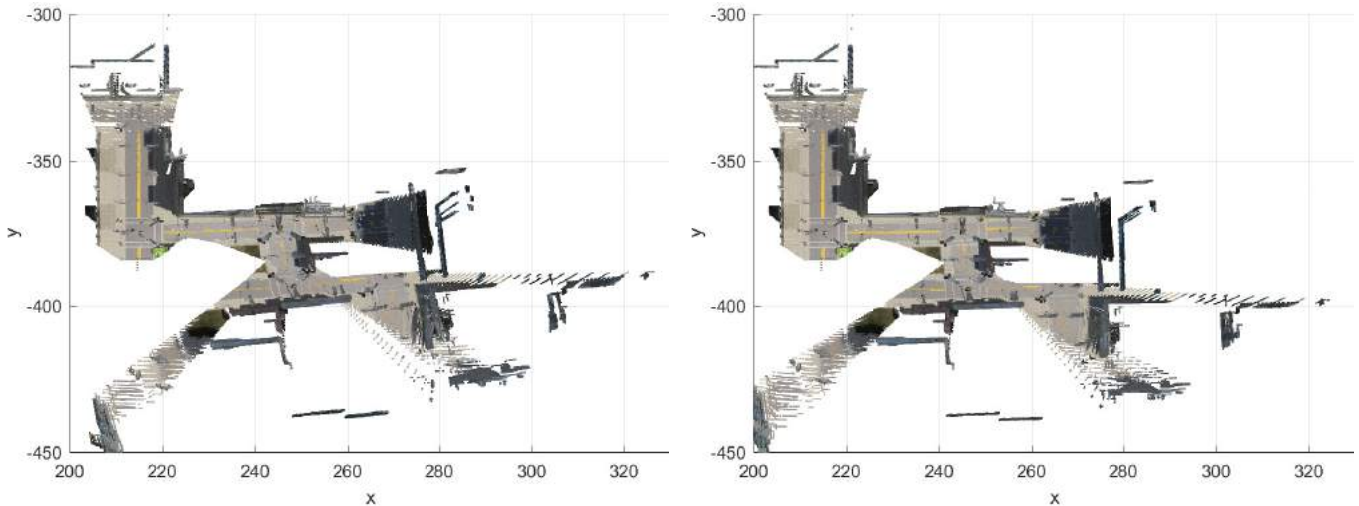


Fig. 6: Top view of the final city map estimate (no loop closure) with no added information (left) vs with planar & angular information added (right). Where some buildings appear to lose orthogonality on the left image, the same buildings appear to show planarity and orthogonality on the right image preserving the global consistency of the map. Noise levels used are as follows: $\Sigma_v = \text{diag}[0.4m, 0.4m, 0.4m, 6^\circ, 6^\circ, 6^\circ]^2$, $\Sigma_w = \text{diag}[0.4m, 0.4m, 0.4m]^2$, $\Sigma_g = [0.01m]^2$ and $\Sigma_r = [1^\circ]^2$

Pre-multiplying by $(K^\perp)^T$ gives:

$$(K^\perp)^T A^T A K^\perp \delta^{*'} = (K^\perp)^T A^T b \quad (11)$$

We solve for $\delta^{*'}$ and then reconstruct $\delta^* = K^\perp \delta^{*'}$ for the solution of the actual system.

Note that due to the sparsity and block structure of the matrix $A^T A$, the product $(K^\perp)^T A^T A K^\perp$ and $(K^\perp)^T A^T b$ can be computed efficiently by exploiting the block structure.

ACKNOWLEDGMENT

This research was supported by the Australian Research Council through the ‘‘Australian Centre of Excellence for Robotic Vision’’ CE140100016. We would like to thank Davide Scaramuzza, Professor, director of the Robotics and Perception Group at University of Zurich, and Juan Adarve, PhD, Australian Centre of Excellence for Robotic Vision, Australian National University for providing the blender-generated city dataset.

REFERENCES

- [1] Sameer Agarwal, Keir Mierle, and Others. Ceres solver. <http://ceres-solver.org>.
- [2] Nicholas Ayache and Olivier D Faugeras. Maintaining representations of the environment of a mobile robot. *IEEE Transactions on Robotics and Automation*, 5(6):804–819, 1989.
- [3] P Cheeseman, R Smith, and M Self. A stochastic map for uncertain spatial relationships. In *4th International Symposium on Robotic Research*, pages 467–474, 1987.
- [4] Paloma de la Puente and Diego Rodríguez-Losada. Feature based graph-slam in structured environments. *Autonomous Robots*, 37(3):243–260, 2014.
- [5] Frank Dellaert and Michael Kaess. Square root sam: Simultaneous localization and mapping via square root information smoothing. *The International Journal of Robotics Research*, 25(12):1181–1203, 2006.
- [6] M.W.M. Gaminí Disssanayake, Paul Newman, Steve Clark, Hugh F Durrant-Whyte, and Michael Csorba. A solution to the simultaneous localization and map building (slam) problem. *IEEE Transactions on Robotics and Automation*, 17(3):229–241, 2001.
- [7] Giorgio Grisetti, Rainer Kummerle, Cyrill Stachniss, and Wolfram Burgard. A tutorial on graph-based slam. *IEEE Intelligent Transportation Systems Magazine*, 2(4):31–43, 2010.
- [8] Giorgio Grisetti, Cyrill Stachniss, Sławomir Grzonka, and Wolfram Burgard. A tree parameterization for efficiently computing maximum likelihood maps using gradient descent. In *Robotics: Science and Systems*, pages 27–30, 2007.
- [9] Simon J Julier and Jeffrey K Uhlmann. A counter example to the theory of simultaneous localization and map building. In *IEEE International Conference on Robotics and Automation (ICRA), Proceedings 2001*, volume 4, pages 4238–4243. IEEE, 2001.
- [10] Michael Kaess. Simultaneous localization and mapping with infinite planes. In *IEEE International Conference on Robotics and Automation (ICRA), 2015*, pages 4605–4611. IEEE, 2015.
- [11] Michael Kaess, Hordur Johannsson, Richard Roberts, Viorela Ila, John J Leonard, and Frank Dellaert. isam2: Incremental smoothing and mapping using the bayes tree. *The International Journal of Robotics Research*, page 0278364911430419, 2011.
- [12] Michael Kaess, Ananth Ranganathan, and Frank Dellaert. isam: Incremental smoothing and mapping. *IEEE Transactions on Robotics*, 24(6):1365–1378, 2008.
- [13] Kurt Konolige, Giorgio Grisetti, Rainer Kummerle, Wolfram Burgard, Benson Limketkai, and Regis Vincent. Efficient sparse pose adjustment for 2d mapping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010*, pages 22–29. IEEE, 2010.
- [14] Rainer Kummerle, Giorgio Grisetti, Hauke Strasdat, Kurt Konolige, and Wolfram Burgard. g2o: A general framework for graph optimization. In *IEEE International Conference on Robotics and Automation (ICRA), 2011*, pages 3607–3613. IEEE, 2011.
- [15] Tae-kyeong Lee, Seungwook Lim, Seongsoo Lee, Shouan An, and Se-young Oh. Indoor mapping using planes extracted from noisy rgb-d sensors. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2012*, pages 1727–1733. IEEE, 2012.
- [16] John J Leonard, Hugh F Durrant-Whyte, and Ingemar J Cox. Dynamic map building for an autonomous mobile robot. *The International Journal of Robotics Research*, 11(4):286–298, 1992.
- [17] Feng Lu and Evangelos Miliotis. Globally consistent range scan alignment for environment mapping. *Autonomous robots*, 4(4):333–349, 1997.
- [18] Mladen Mazuran, Gian Diego Tipaldi, Luciano Spinello, Wolfram Burgard, and Cyrill Stachniss. A statistical measure for map consistency in slam. In *IEEE International Conference on Robotics and Automation (ICRA), 2014*, pages 3650–3655. IEEE, 2014.
- [19] Lukas Polok, Viorela Ila, Marek Solony, Pavel Smrz, and Pavel Zemcik. Incremental block cholesky factorization for nonlinear least squares in robotics. In *Proceedings of Robotics: Science and Systems*, Berlin, Germany, June 2013.
- [20] Lukas Polok, Marek Solony, Viorela Ila, Pavel Smrz, and Pavel Zemcik. Efficient implementation for block matrix operations for nonlinear least squares problems in robotic applications. In *IEEE International Conference on Robotics and Automation (ICRA), 2013*, pages 2263–2269. IEEE, 2013.
- [21] Randall Smith, Matthew Self, and Peter Cheeseman. Estimating uncertain spatial relationships in robotics. In *Autonomous robot vehicles*, pages 167–193. Springer, 1990.
- [22] Gilbert Strang. *Linear algebra and its applications*. Number 04; QA184, S8. 1976.
- [23] Julian Straub, Guy Rosman, Oren Freifeld, John J Leonard, and John W Fisher. A mixture of manhattan frames: Beyond the manhattan world. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014*, pages 3770–3777. IEEE, 2014.
- [24] Richard Szeliski and Philip HS Torr. Geometrically constrained structure from motion: Points on planes. In *European Workshop on 3D Structure from Multiple Images of Large-Scale Environments*, pages 171–186. Springer, 1998.
- [25] Yuichi Taguchi, Yong-Dian Jian, Srikumar Ramalingam, and Chen Feng. Point-plane slam for hand-held 3d sensors. In *IEEE International Conference on Robotics and Automation (ICRA), 2013*, pages 5182–5189. IEEE, 2013.
- [26] Keisuke Tateno, Federico Tombari, and Nassir Navab. Real-time and scalable incremental segmentation on dense slam. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015*, pages 4465–4472. IEEE, 2015.
- [27] Alexander JB Trevor, John G Rogers, and Henrik I Christensen. Planar surface slam with 3d and 2d sensors. In *IEEE International Conference on Robotics and Automation (ICRA), 2012*, pages 3041–3048. IEEE, 2012.
- [28] Rudolph Triebel and Wolfram Burgard. Improving simultaneous mapping and localization in 3d using global constraints. In *Proceedings of the National Conference on Artificial Intelligence*, volume 20, page 1330. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2005.
- [29] Zichao Zhang, Henri Rebecq, Christian Forster, and Davide Scaramuzza. Benefit of large field-of-view cameras for visual odometry. In *IEEE International Conference on Robotics and Automation (ICRA), 2016*. IEEE, 2016.
- [30] Marco Zucchelli, José Santos-Victor, and Henrik I Christensen. Constrained structure and motion estimation from optical flow. In *16th International Conference on Pattern Recognition, Proceedings 2002.*, volume 1, pages 339–342. IEEE, 2002.