Sharing Heterogeneous Spatial Knowledge: Map Fusion between Asynchronous Monocular Vision and Lidar or Other Prior Inputs

Yan Lu, Joseph Lee, Shu-Hao Yeh, Hsin-Min Cheng, Baifan Chen, and Dezhen Song *

Abstract To enable low-cost mobile devices and robots equipped with monocular cameras to obtain accurate position information in GPS-denied environments, we propose to use pre-collected lidar or other prior data to rectify imprecise visual simultaneous localization and mapping (SLAM) results. This leads to a novel and nontrivial problem that fuses vision and prior/lidar data acquired at different perspectives and time. In fact, the lidar inputs can be replaced by other prior mapping inputs as long as we can extract vertical planes from these inputs. Hence, they are referred as prior/lidar data in general. We exploit the planar structure extracted from both vision and prior/lidar data and use it as the anchoring information to fuse the heterogeneous maps. We formulate a constrained global bundle adjustment using coplanarity constraints and solve it using a penalty-barrier approach. By error analysis we prove that the coplanarity constraints help reduce the estimation uncertainties. We have implemented the system and tested it with real data. The initial results show that our algorithm significantly reduces the absolute trajectory error of visual SLAM by as much as 68.3%.

Yan Lu
Honda Research Institute USA, Mountain View, California, USA, e-mail: sinoluyan@gmail.com

Joseph Lee
U.S. Army TARDEC, Warren, Michigan, USA, e-mail: joseph.s.lee34.civ@mail.mil

Shu-Hao Yeh · Hsin-Min Cheng · Dezhen Song
Texas A&M University, College Station, Texas, USA, e-mail: ericex1015@tamu.edu, dora90474@gmail.com, dzsong@cse.tamu.edu

Baifan Chen
Central South University, Hunan, China, e-mail: chenbaifan@csu.edu.cn

* This work was supported in part by National Science Foundation under NRI-1426752, NRI-1526200 and NRI-1748161, and in part by National Science Foundation of China under 61403423.

1 Introduction

Since GPS signals are often challenged in indoor or urban environments, many researchers focus on developing simultaneous localization and mapping (SLAM) algorithms using onboard sensors for robots or mobile devices. Nowadays regular cameras are the most available and inexpensive sensor which relies on visual SLAM algorithms. RGB-D cameras become more available but are unreliable in strong sunlight. A lidar is often considered as the most reliable mapping sensor, but it is too power hungry, bulky, and expensive for many mobile robots or devices. A mobile device or a small robot often can only rely on a monocular camera due to power and size constraints.

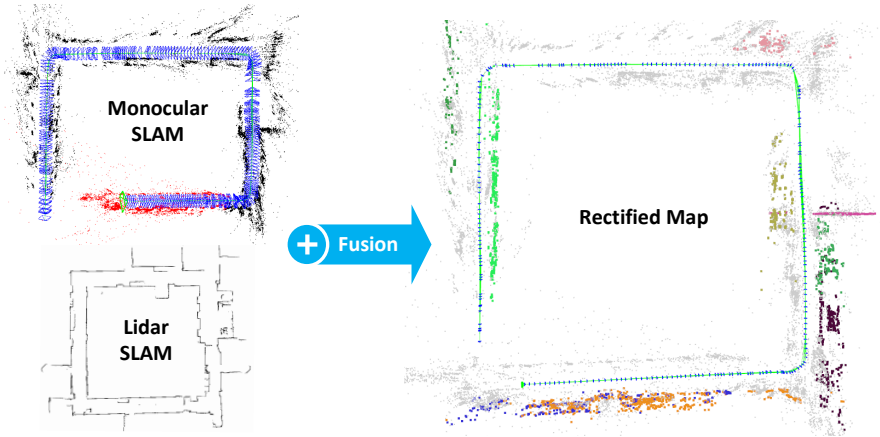


Fig. 1: The inputs of our map fusion include a low-quality 3D map produced by a monocular visual SLAM, and a high-precision prior map generated by lidar SLAM other methods. The map fusion rectifies the 3D map by leveraging vertical planes commonly available in both maps and outputs a more accurate 3D map.

However, monocular visual SLAM algorithms often generate low-quality maps due to scale and angular drift, as illustrated in Figure 1. To address these problems, we propose to pre-scan the environment with a high-precision 2D or 3D lidar. The lidar generated map (see Figure 1) will be shared among mobile devices or other robots with only monocular cameras. If we can rectify the low-quality visual SLAM map to the lidar-level accuracy (see Figure 1), mobile device users and robots will enjoy high localization accuracy by just using low-cost cameras, which has a large advantage in many applications. In fact, we can also take other prior maps as inputs to rectify visual SLAM results instead of lidar inputs. The only requirement is that we can extract vertical planes from these prior mapping inputs. For example, Google maps™ often contains building exterior planes.

Since lidar point clouds or prior maps and image feature points cannot be directly registered to each other, this requires us to fuse the spatial knowledge (i.e. landmarks of maps) asynchronously captured from different perspectives and platforms by the heterogeneous sensors, which presents a new research problem. We detect high-level landmarks such as vertical planes that can be identified in both sensing modalities. We formulate a constrained global bundle adjustment using coplanarity constraints and solve the problem using a penalty-barrier approach. By error analysis we prove that the coplanarity constraints help reduce the estimation uncertainties. We have implemented the system and tested it with real data. The initial results show that our algorithm significantly reduces the absolute trajectory error of visual SLAM by as much as 68.3%.

2 RELATED WORK

The map fusion problem can be viewed as a post processing step in visual SLAM, which is a fast developing area. More specifically, this paper builds upon recent works including optimization techniques, different sensor and feature configurations, and collaborative map merging for multiple robots.

An SLAM algorithm simultaneously estimates camera/robot poses and landmark positions, which is a fundamental problem in robotics [33] and computer vision. A full fledged SLAM framework consists of subproblems such as tracking, mapping, and loop closing. For efficiency, recent systems [17, 31, 29, 9, 23] implement a front-end and back-end system running in parallel, where the front-end performs tracking in real-time while the back-end refines both the resulting trajectory and the map, and closes loops occasionally. To increase the speed of the back-end, different optimization techniques are proposed. Recent optimization methods take advantage of sparse matrix structures in the SLAM problem. In [20], an implementation of a hypergraph-based sparse non-linear optimization framework called g2o is presented. An optimization method known as iSAM2 [16] uses a Bayes tree to incrementally update the sparse matrix for an on-line SLAM system. Building on the g2o framework to exploit sparse matrix computation, our proposed map fusion can be viewed as a back-end system which runs asynchronously from a front-end tracking and mapping process.

SLAM can be performed with different exteroceptive sensors or their combinations including regular cameras, lidars, and RGB-D cameras. Regular camera-based SLAM is referred as visual SLAM, where two main approaches exist: filtering and structure-from-motion (SFM) using an optimization approach. For filtering, the extended Kalman filters (EKF) or its variants have been used extensively [7, 26, 5], whereas the dominating method for SFM is bundle adjustment (BA) [27, 19]. The BA method is an optimization-based method which minimizes the reprojection error over multiple image frames. Strasdat et al. [30] have compared EKF against BA and pointed out that unless in high uncertainty situations, BA has a better performance and accuracy than EKF. Our proposed method uses a batch optimization to estimate

the fused map. However, monocular vision-based SLAM is the most difficult problem due to the infamous scale drifting issue. Stereo vision can relieve the issue but is often limited by the baseline and power constraints of small devices.

Depth sensors such as RGB-D cameras or lidars [14, 8, 24] can help address the scale drifting issue in monocular vision. While RGB-D cameras are widely used for indoor environments, lidars are still the most favorable sensor due to longer sensing range, wide field of view, and robustness to lighting conditions, making it suitable for both indoor and outdoor applications [12]. Kohlbrecher et al. [18] register 2D lidar scans to a 2D occupancy grid map based on an image registration technique. Zhang and Singh [34] use a 2-axis lidar and develop a real-time 3D mapping method with low-drift by separating the odometry and mapping task. In these methods, the lidar map is accurate enough so that post-processing is not required.

Combining both vision and lidar inputs to utilize the benefit of each sensor has been proposed as well. Newman et al. [25] employ a 3D lidar to map buildings and use vision to detect loop closure. Zhang and Singh [35] use vision to handle rapid motion while the lidar warrants low-drift and robustness to lighting changes. However, these methods require that the vision and lidar data are synchronized and captured by the same robot. The resulting map is also limited to the hosting robot for the localization usage. Our method is intended to fuse the vision and lidar maps acquired by different robots and generate maps that can help different cameras or mobile device users. Caselitz et al. [4] study localizing a monocular camera within a lidar map, but do not investigate the map fusion problem.

Map merging has been studied for multi-robot systems in indoor environments. Dedeoglu and Sukhatme [6] combine topological maps using landmarks detected by sonar, vision, and laser. Fox et al. [11] merge lidar maps with robots actively detecting each other to estimate their relative positions. Carpin [3] proposes a map merging technique based on Hough transforms to merge occupancy grid maps. Baudouin et al. [1] propose a method that merges robot paths with different scales generated by multi-modal vision sensors such as perspective, fish-eye, or omnidirectional cameras. These prior works shed light on our problem but they focus on map building with the same type of sensors.

In a lidar map or other prior map, the most visible features are wall planes. Many studies make use of the fact that indoor/urban environments have planar structures [13, 2, 22]. Lu and Song introduce a multilayer feature graph (MFG) [21] for a visual SLAM approach using heterogeneous landmarks including planes. Taguchi et al. [32] use an RGB-D camera and propose a method that uses combinations of primitives, i.e. points and planes, for faster pose estimation. More recently, Salas-Moreno et al. [28] have proposed a dense planar SLAM using an RGB-D camera.

3 Problem Definition

A robot equipped with a monocular camera navigates in an area for which a lidar map or other prior map is given. From the prior/lidar map $\mathscr{M}_L$, a set of line segments

are extracted to represent major vertical planes from a top down perspective. We assume 1) the camera is calibrated, and 2) the vertical direction is known which can be obtained through either analyzing vanishing points from images or sensing gravity direction from inertial sensors.

Let the camera pose at time $k \in \mathbb{N}$ be $\mathrm{T}_k \in SE(3)$. Define $\mathscr{T}_k = \{\mathrm{T}_1, \cdots, \mathrm{T}_k\}$ as the collection of camera poses up to time k. Let $\mathscr{M}_V$ denote the 3D map built by visual SLAM, which consists of a set of 3D points $\mathbf{p}_i = [x_i, y_i, z_i]^T \in \mathbb{R}^3$.

The problem is: Given $\mathscr{M}_V$ and $\mathscr{T}_k$ up to time k, rectify $\mathscr{M}_V$ and $\mathscr{T}_k$ with $\mathscr{M}_L$.

4 Map Fusion and Uncertainty Analysis

The map built by visual SLAM is inevitably subject to drift. To rectify it, we periodically detect vertical planes from $\mathscr{M}_V$ and associate them with those in $\mathscr{M}_L$. We then incorporate the plane information from $\mathscr{M}_L$ as additional constraints into the bundle adjustment of visual SLAM to rectify $\mathscr{M}_V$ and $\mathscr{T}_k$. Next we first present our map fusion algorithm including plane detection and bundle adjustment, and then perform uncertainty analysis to show the benefit of including additional plane information in visual SLAM.

4.1 Map Fusion

4.1.1 Plane Detection and Matching

In visual SLAM we periodically detect vertical planes from 3D points using sequential RANSAC [10]. To keep the problem manageable, we only consider a fixed number of recent points that are visible in at least three adjacent keyframes. At each iteration of RANSAC, we randomly pick three points from the pool and compute a candidate plane. As we know the vertical direction, we can filter out candidate planes in the first place. Then we check how many points support this candidate plane by computing a consensus score for each point in the pool. If enough support points are found, we add the plane to $\mathscr{M}_V$ and remove the support points from the pool. The RANSAC process keeps going until the maximum number of iterations is reached.

In this process, the consensus score is key to properly evaluate whether a point supports a candidate plane. An intuitive way is to compute the perpendicular point-to-plane distance in 3D as the consensus score. However, the map is only up-to-scale in monocular SLAM, which poses difficulty to choosing a proper threshold for the scores. Here we propose a novel consensus score in image space. Given a 3D point $\mathbf{p}$ and a 3D plane π, suppose $\mathbf{p}$ is observed by camera at times $k \in \mathscr{K}$. As illustrated in Figure 2, we first find the point $\mathbf{p}'$ lying on π that is closest to $\mathbf{p}$, then define

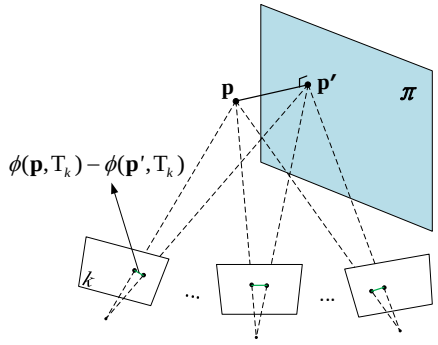


Fig. 2 Consensus score for evaluating whether a 3D point supports a plane. Defined in image space, it avoids the scale ambiguity in monocular visual SLAM.

$$d_{IMG}(\mathbf{p},\pi) = \max_{k\in\mathscr{K}} \|\phi(\mathbf{p},\mathrm{T}_k) - \phi(\mathbf{p}',\mathrm{T}_k)\|, \tag{1}$$

where $\phi(\cdot,\cdot)$ is the camera projection function. Defined in image space, this metric is independent of the absolute map scale and eases the task of setting thresholds.

Once a vertical plane is detected from $\mathscr{M}_V$, we need to match it against $\mathscr{M}_L$. This problem can be solved by topology graph matching, or using manual inputs if the number of planes is small. We ignore planes in $\mathscr{M}_V$ that do not correspond to a physical plane, which can be achieved by detecting appearance consistency.

4.1.2 Optimization

Since prior/lidar map $\mathscr{M}_L$ is much more accurate than $\mathscr{M}_V$, we assume $\mathscr{M}_L$ to be error-free. Our goal is to register $\mathscr{M}_V$ against $\mathscr{M}_L$ so that the mapping error in $\mathscr{M}_V$ is minimized. This is achieved by leveraging plane correspondences. Suppose by time k we have found correspondences for planes $\pi_j, j \in J_k$. We formulate the following optimization problem to rectify $\mathscr{M}_V$ and $\mathscr{T}_k$.

$$\begin{aligned} \min_{\mathscr{M}_V,\mathscr{T}_k} \sum_{\kappa=1}^{k} \sum_{i\in\mathscr{I}(\kappa)} \|\phi(\mathbf{p}_i,\mathrm{T}_\kappa) - \mathbf{m}_{i,\kappa}\|^2_{\Sigma_{i,\kappa}} \\ \text{s.t. } d_\perp(\mathbf{p},\pi_j) = 0,\ \forall \mathbf{p}\in\pi_j, j\in J_k \end{aligned} \tag{2}$$

where $\mathscr{I}(\kappa)$ collects all indexes of points visible by camera at time κ, $\mathbf{m}_{i,\kappa}$ is the image observation for $\mathbf{p}_i$ at time κ, $\Sigma_{i,\kappa}$ is the covariance of the measurement noise of $\mathbf{m}_{i,\kappa}$, $\|\cdot\|_\Sigma$ denotes Mahalanobis distance, and $d_\perp(\cdot,\cdot)$ represents the perpendicular Euclidean distance between a point and a plane in 3D. Here we abuse the notation $\mathbf{p}\in\pi_j$ to indicate the relation that $\mathbf{p}$ is supposed to reside on plane π_j. The exact position of π_j in (2) is retrieved from $\mathscr{M}_L$.

Eq. (2) is a constrained optimization problem. To solve it, we convert it to an unconstrained optimization problem by adding a penalty function as follows

$$\min_{\mathscr{M}_V,\mathscr{T}_k} \sum_{\kappa=1}^{k} \sum_{i\in\mathscr{I}(\kappa)} \|\phi(\mathbf{p}_i,\mathrm{T}_\kappa)-\mathbf{m}_{i,\kappa}\|^2_{\Sigma_{i,\kappa}} + w\sum_{j\in J_k}\sum_{\mathbf{p}\in\pi_j} d_\perp(\mathbf{p},\pi_j)^2 \tag{3}$$

The first term is the same re-projection errors as in (2), and the second term is a penalty term. We first solve (3) with a relatively small weight w, and use the solution as initial point to solve (3) again with an increased w. We repeat this process until the change in the solution is negligible. It is worth noting that we use $d_\perp(\cdot,\cdot)$ instead of $d_{IMG}(\cdot,\cdot)$ in (2) and (3) because the scale ambiguity does not cause problems in optimization.

4.2 Uncertainty Analysis

In this section we show the effect of incorporating known plane information in visual SLAM by analyzing the estimation uncertainties. To simplify the analysis, we assume the first two camera poses are given, which essentially fixes the absolute scale for visual SLAM. Upon time k, we want to estimate all $k-2$ camera poses and all n map points. Let $\mathbf{y} := \left[\mathrm{T}_3^T,\cdots,\mathrm{T}_k^T\right]^T$ be the vector comprising camera poses, where T_k is minimally parameterized. Without loss of generality, we assume that the last m points are from a plane π. Furthermore, we choose a world coordinate system such that its X-Y plane is parallel to π. This leads to the following coplanarity constraint.

Definition 1 (Coplanarity). Points $\{\mathbf{p}_i | i = n-m+1,\cdots,n\}$ reside on a known 3D plane π, which is parallel to the X-Y plane. Thus the Z-coordinates of all points on π are a constant, denoted by z_π.

We next analyze the estimation uncertainties of the camera poses and points without and with using the coplanarity constraint, respectively.

4.2.1 No Coplanarity

In this case, we do not consider the coplanarity in Definition 1 by treating each point as independent. Let $\mathbf{z}_1 := \left[\mathbf{p}_1^T,\cdots,\mathbf{p}_{n-m}^T\right]^T$ comprise points not residing on π, and $\mathbf{z}_2 := \left[\mathbf{p}_{n-m+1}^T,\cdots,\mathbf{p}_n^T\right]^T$ comprise points on π. We reorder the elements of $\mathbf{z}_2$ such that

$$\mathbf{z}_2 = [\underbrace{x_{n-m+1},y_{n-m+1},\cdots,x_n,y_n}_{:=\mathbf{z}_{XY}^T},\ \underbrace{z_{n-m+1},\cdots,z_n}_{:=\mathbf{z}_Z^T}]^T = \left[\mathbf{z}_{XY}^T,\mathbf{z}_Z^T\right]^T.$$

Let $\mathbf{x} = \left[\mathbf{y}^T,\mathbf{z}_1^T,\mathbf{z}_2^T\right]^T = [\underbrace{\mathbf{y}^T,\mathbf{z}_1^T,\mathbf{z}_{XY}^T}_{:=\mathbf{x}_1^T},\mathbf{z}_Z^T]^T = \left[\mathbf{x}_1^T,\mathbf{z}_Z^T\right]^T$. Define a measurement error function

$$\Phi(\mathbf{x}) = \begin{bmatrix} \vdots \\ \phi(\mathbf{p}_i, \mathrm{T}_\kappa) - \mathbf{m}_{i,\kappa} \\ \vdots \end{bmatrix} \quad (4)$$
$$\forall \kappa = 1, \cdots, k, \forall i \in \mathscr{I}(\kappa).$$

Recall that $\mathbf{m}_{i,\kappa}$ is the image observation of $\mathbf{p}_i$ at time κ, and $\mathscr{I}(\kappa)$ collects the point indexes visible at time κ.

We estimate $\mathbf{x}$ by

$$\min_{\mathbf{x}} \Phi(\mathbf{x})^T \Sigma^{-1} \Phi(\mathbf{x}) \quad (5)$$

where $\Sigma = diag(\cdots, \Sigma_{i,\kappa}, \cdots)$, with $\Sigma_{i,\kappa}$ being the covariance of $\mathbf{m}_{i,\kappa}$.

Lemma 1. *Let* $\mathbf{x}^* = \left[\mathbf{x}_1^{*T}, \mathbf{z}_Z^{*T}\right]^T$ *be the solution of the problem in (5). Under Gaussian noise assumption, the first order approximation of the covariance of* $\mathbf{x}_1^*$ *is*

$$\mathtt{Cov}(\mathbf{x}_1^*) = \left(\mathbf{J}_{\mathbf{x}_1}^T \Sigma^{-1} \mathbf{J}_{\mathbf{x}_1} - \mathbf{J}_{\mathbf{x}_1}^T \Sigma^{-1} \mathbf{J}_{\mathbf{z}_Z} \left(\mathbf{J}_{\mathbf{z}_Z}^T \Sigma^{-1} \mathbf{J}_{\mathbf{z}_Z}\right)^{-1} \mathbf{J}_{\mathbf{z}_Z}^T \Sigma^{-1} \mathbf{J}_{\mathbf{x}_1}\right)^{-1} \quad (6)$$

where $\mathbf{J}_{\mathbf{x}_1} = \left.\frac{\partial \Phi}{\partial \mathbf{x}_1}\right|_{\mathbf{x}_1 = \mathbf{x}_1^*}$, $\mathbf{J}_{\mathbf{z}_Z} = \left.\frac{\partial \Phi}{\partial \mathbf{z}_Z}\right|_{\mathbf{z}_Z = \mathbf{z}_Z^*}$.

4.2.2 Apply Coplanarity

In this case, we explicitly model coplanarity in Definition 1. As a result, $\forall i \in [n-m+1, \cdots, n]$, we have $\mathbf{p}_i = [x_i, y_i, z_\pi]^T$, and thus $\phi(\mathbf{p}_i, \mathrm{T}_\kappa) = \phi\left([x_i, y_i, z_\pi]^T, \mathrm{T}_\kappa\right)$. Recall $\mathbf{x}_1 = \left[\mathbf{y}^T, \mathbf{z}_1^T, \mathbf{z}_{XY}^T\right]^T$. In this case, $\mathbf{x}_1$ consists of all the parameters we need to estimate. Define a measurement error function

$$\Phi'(\mathbf{x}_1) = \begin{bmatrix} \vdots \\ \phi(\mathbf{p}_i, \mathrm{T}_\kappa) - \mathbf{m}_{i,\kappa} \\ \vdots \end{bmatrix} \quad (7)$$
$$\forall \kappa = 1, \cdots, k, \forall i \in \mathscr{I}(\kappa).$$

We estimate $\mathbf{x}_1$ by

$$\min_{\mathbf{x}_1} \Phi'(\mathbf{x}_1)^T \Sigma^{-1} \Phi'(\mathbf{x}_1). \quad (8)$$

Lemma 2. *Let* $\mathbf{x}_1'^*$ *be the solution of the problem in (8). Under Gaussian noise assumption, the first order approximation of the covariance of* $\mathbf{x}_1'^*$ *is*

$$\mathtt{Cov}(\mathbf{x}'^{*}_1) = \left(\mathbf{J}'^{T}_{\mathbf{x}_1}\Sigma^{-1}\mathbf{J}'_{\mathbf{x}_1}\right)^{-1} \tag{9}$$

where $\mathbf{J}'_{\mathbf{x}_1} = \frac{\partial \Phi}{\partial \mathbf{x}_1}\Big|_{\mathbf{x}_1=\mathbf{x}'^{*}_1}$.

4.2.3 Uncertainty Reduction

Now we have estimated $\mathbf{x}_1$ under two scenarios and its respective covariances $\mathtt{Cov}(\mathbf{x}^{*}_1)$ and $\mathtt{Cov}(\mathbf{x}'^{*}_1)$. The following theorem reveals how the coplanarity constraint affects the estimation uncertainty.

Theorem 1. *The coplanarity in Definition 1 reduces the estimation uncertainty of the parameters in* $\mathbf{x}_1$. *Specifically,*

$$\lambda_j\left(\mathtt{Cov}\left(\mathbf{x}'^{*}_1\right)\right) < \lambda_j\left(\mathtt{Cov}\left(\mathbf{x}^{*}_1\right)\right),\ 0 \le j \le len(\mathbf{x}_1) \tag{10}$$

where $\lambda_j(\cdot)$ *denotes the j-th largest eigenvalue,* $len(\cdot)$ *is the length of a vector, and* $\mathbf{x}'^{*}_1$ *and* $\mathbf{x}^{*}_1$ *are the optimal estimation of* $\mathbf{x}_1$ *with and without knowing the coplanarity, respectively.*

Proof. Let us write $M_1 \succ M_2$ if matrices M_1 and M_2 are real symmetric and $M_1 - M_2$ is positive definite.

As $\mathbf{x}^{*}$ and $\mathbf{x}'^{*}_1$ are the global optimal solution to (5) and (8), respectively, it is easy to see $\mathbf{x}^{*}_1 = \mathbf{x}'^{*}_1$, and $\mathbf{z}^{*}_Z = [z_\pi, \cdots, z_\pi]^T$. Consequently, $\mathbf{J}_{\mathbf{x}_1} = \mathbf{J}'_{\mathbf{x}_1}$. From Lemma 1 and Lemma 2, we derive that

$$\mathtt{Cov}\left(\mathbf{x}'^{*}_1\right)^{-1} - \mathtt{Cov}\left(\mathbf{x}^{*}_1\right)^{-1} = \mathbf{J}^{T}_{\mathbf{x}_1}\Sigma^{-1}\mathbf{J}_{\mathbf{z}_Z}\left(\mathbf{J}^{T}_{\mathbf{z}_Z}\Sigma^{-1}\mathbf{J}_{\mathbf{z}_Z}\right)^{-1}\mathbf{J}^{T}_{\mathbf{z}_Z}\Sigma^{-1}\mathbf{J}_{\mathbf{x}_1}.$$

Since we use minimal parameterization for $\mathbf{x}_1$ and $\mathbf{z}_Z$, it holds that

$$\mathbf{J}^{T}_{\mathbf{x}_1}\Sigma^{-1}\mathbf{J}_{\mathbf{z}_Z}\left(\mathbf{J}^{T}_{\mathbf{z}_Z}\Sigma^{-1}\mathbf{J}_{\mathbf{z}_Z}\right)^{-1}\mathbf{J}^{T}_{\mathbf{z}_Z}\Sigma^{-1}\mathbf{J}_{\mathbf{x}_1} \succ 0,$$

which leads to $\mathtt{Cov}\left(\mathbf{x}'^{*}_1\right)^{-1} \succ \mathtt{Cov}\left(\mathbf{x}^{*}_1\right)^{-1}$. According to Theorem 7.7.3 in [15], it holds

$$\mathtt{Cov}\left(\mathbf{x}'^{*}_1\right)^{-1} \succ \mathtt{Cov}\left(\mathbf{x}^{*}_1\right)^{-1} \Leftrightarrow \mathtt{Cov}\left(\mathbf{x}^{*}_1\right) \succ \mathtt{Cov}\left(\mathbf{x}'^{*}_1\right),$$

which further leads to (10) per Corollary 7.7.4 in [15]. □

It is worth noting that Theorem 1 can be easily generalized when the plane is not parallel to X-Y plane, but an arbitrary plane. Furthermore, when more planes are known, it is intuitive to see that the estimation uncertainties can be further reduced.

5 Experiments

In this section, we first verify Theorem 1 by simulated experiments, and then evaluate our proposed method using real world data.

5.1 Simulation

In this simulation the robot navigates through an L-shaped corridor as illustrated in Figure 3(a). The corridor consists of 4 walls: Plane A, B, C, and D, and each wall has around 50 randomly distributed points. The robot starts at the origin, first moves forward and then turns right at the corner. The image resolution is 640×480 and the camera focal length is 500. As assumed in Section 4.2 the first two camera poses are given.

We first compute the uncertainties using (6) which takes no coplanarity information into account. Figure 3(b) shows the uncertainties of all camera positions and 3D points. As expected the uncertainties gradually increase as the robot moves forward.

Now consider the constraint that a set of points reside on Plane B whose position is known. The uncertainties of map points and camera poses are computed using (9) and illustrated in Figure 3(c). Compared with Figure 3(b), it is obvious that all the uncertainties are reduced.

To illustrate how the plane position affects the uncertainty reduction, we consider the coplanarity of Plane C instead and show the resulted uncertainties in Figure 3(d). Compared with Figure 3(b), it is also obvious that the uncertainties are reduced. However, by comparing Figure 3(c) and Figure 3(d), we see the different effects. Figure 4 further compares the determinants of all camera pose covariances for the three scenarios described above.

5.2 Real World Test

We have implemented our system based on ORB-SLAM [23], a state-of-the-art monocular visual SLAM system, though our method is applicable to any other monocular visual SLAM. We evaluate our system on one indoor and two outdoor datasets, as described below.

- *HRBB4*: This is a sequence of 12,000 images collected in an office corridor environment with the ground truth of camera trajectory provided [21]. To evaluate our method, we collected lidar data using a Hokuyo (UTM-30LX) in the same environment two years after the images were acquired. We generate a lidar map using Hector-SLAM [18] and extract major line segments (i.e. vertical planes) as illustrated in Figure 5(a).

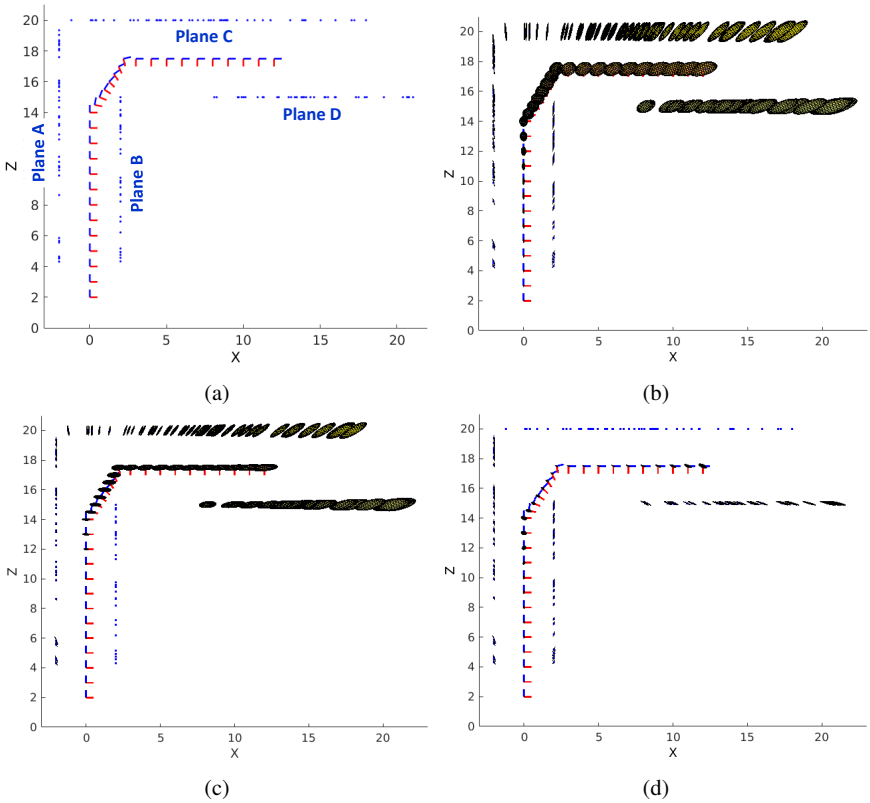


Fig. 3: (a) The simulation setup. An L-shaped corridor comprises four walls, Plane A, B, C and D, each consisting of around 50 points. The camera starts from the origin, moves forward first and turns right at the corner. (b-d) The uncertainties of camera positions and points in the case of (b) no coplanarity being considered, (c) the coplanarity constraint for Plane B being applied, and (d) the coplanarity constraint for Plane C being applied.

- *KITTI-07*: This is an outdoor image sequence from the KITTI dataset [12], consisting of 1101 images recorded by an autonomous driving platform. The ground truth of camera trajectory is provided by a high-grade GPS-INS system. We generate a 3D lidar map from the accompanied velodyne data using LOAM [34] and treat it as a 2D map by projecting it to the ground plane as illustrated in Figure 6(a).
- *KITTI-00*: This is another outdoor image sequence from the KITTI dataset [12], consisting of 4541 images. We use Google satellite map of the corresponding area as our prior map, as shown in Figure 7(a).

To evaluate the SLAM accuracy, we compute the statistics of absolute trajectory errors (ATE), including RMS (root-mean-square) error, SD (standard devia-

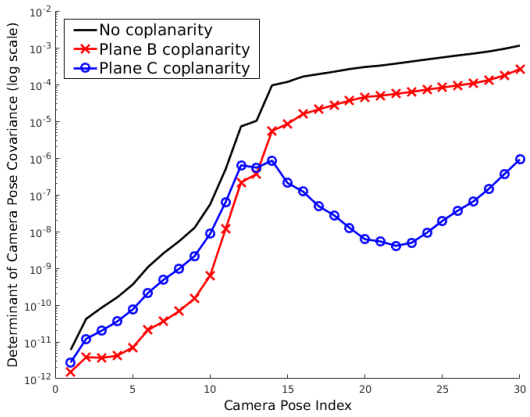
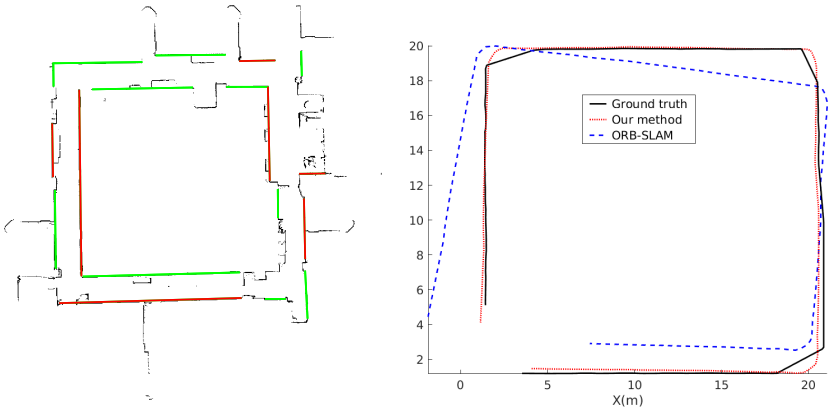


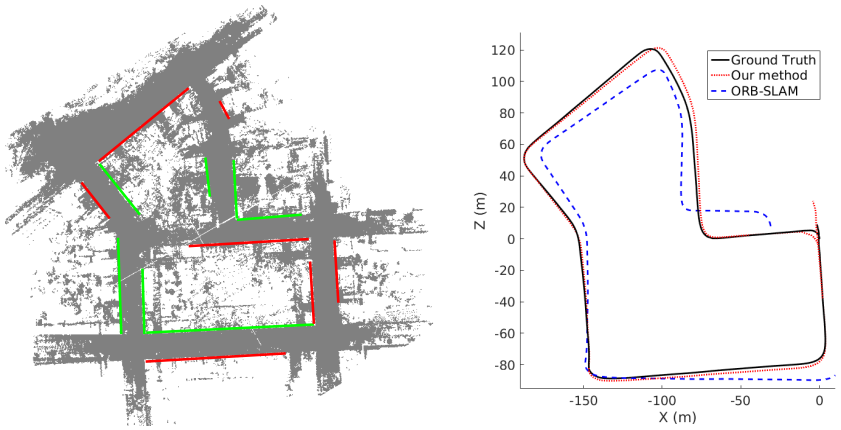
Fig. 4 Determinants of camera pose covariances under different scenarios. Note how the coplanarity reduces the uncertainties. Note that the vertical axis uses log scale.



(a) (b)

Fig. 5: (a) Vertical planes (red and green lines) overlaid on the HRBB4 lidar map (black). Red lines indicate the planes that are matched and thus used for map fusion. (b) Camera trajectory ground truth and estimates for the HRBB4 dataset.

tion), Max(maximum), and the ratio of RMS error over the trajectory length for my method and ORB-SLAM. Note that loop closing is disabled in ORB-SLAM for fair comparison. In Table 1 we see that our algorithm significantly reduces the ATE by leveraging the prior 2D maps. The estimated camera trajectories versus the ground truth are shown in Figures 5(b), 6(b), and 7(b). Compared with ORB-SLAM, our method reduces the RMS error by 60.6% for the HRBB4 dataset, by 69.8% for the KITTI-07 dataset, and by 74.4% for the KITTI-00 dataset. On average our method reduces the RMS error by 68.3%.



(a) (b)

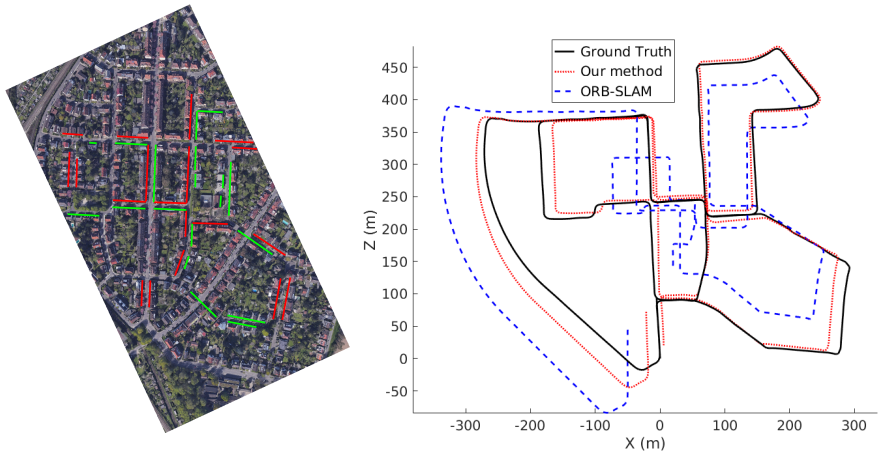
Fig. 6: (a) Vertical planes (red and green lines) overlaid on the KITTI-07 lidar map (gray). Red lines indicate the planes that are matched and thus used for map fusion. (b) Camera trajectory ground truth and estimates for the KITTI-07 dataset.

Table 1: Absolute Trajectory Errors (m)

Dataset	ORB-SLAM				Our method			
	RMS(m)	SD(m)	Max(m)	RMS/Traj	RMS(m)	SD(m)	Max(m)	RMS/Traj
HRBB4	1.32	0.68	3.2	1.89%	0.52	0.27	1.37	0.74%
KITTI-07	15.06	8.61	36.99	2.17%	4.55	2.82	15.04	0.65%
KITTI-00	62.08	30.04	144.68	1.67%	15.92	8.04	34.04	0.43%

6 Conclusions

We presented a method that fuses two maps generated from different sensory modalities, i.e. monocular vision and prior/lidar data, to assist low-cost devices/robots to obtain high quality localization information. The prior/lidar data can be either maps constructed from lidar inputs or other prior map inputs as along as we can extract vertical planes from them. We exploited the planar structure extracted from both vision and prior/lidar data and use it as the anchoring information to fuse the heterogeneous maps. We formulated a constrained nonlinear optimization problem under global bundle adjustment framework using coplanarity constraints. We solved the problem using a penalty-barrier approach. By error analysis we proved that the coplanarity constraints help reduce the estimation uncertainties. We implemented the system and tested it with real data. The results showed that our algorithm significantly reduced the absolute trajectory error of visual SLAM by as much as 68.3%.



(a) (b)

Fig. 7: (a) Vertical planes (red and green lines) overlaid on the Google satellite map for the KITTI-00 dataset. Red lines indicate the planes that are matched and thus used for map fusion. (b) Camera trajectory ground truth and estimates for the KITTI-00 dataset.

For future work, we will test our method on different indoor environments. One shortcoming of the proposed method is the reliance on vertical plane distribution. Although most indoor environments are dominated by vertical planes, this method would fail in an environment with very few vertical planes. We are interested in extracting other signatures such as object boundary points to handle the problem. We will also explore distributed implementations for the optimization framework. Finally, we are interested in developing cloud-based schemes to further allow algorithms and positional information to be shared among low-cost devices.

References

1. Léo Baudouin, Youcef Mezouar, Omar Ait-Aider, and Helder Araújo. Multi-modal sensors path merging. In *Intelligent Autonomous Systems 13*, pages 191–201. Springer, 2016.
2. András Bódis-Szomorú, Hayko Riemenschneider, and Luc Van Gool. Fast, approximate piecewise-planar modeling based on sparse structure-from-motion and superpixels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 469–476, 2014.
3. Stefano Carpin. Merging maps via hough transform. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 1878–1883. IEEE, 2008.
4. Tim Caselitz, Bastian Steder, Michael Ruhnke, and Wolfram Burgard. Monocular camera localization in 3d lidar maps. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.

5. Javier Civera, Oscar G Grasa, Andrew J Davison, and JMM Montiel. 1-point RANSAC for extended Kalman filtering: Application to real-time structure from motion and visual odometry. *Journal of Field Robotics*, 27(5):609–631, 2010.
6. Göksel Dedeoglu and Gaurav S Sukhatme. Landmark-based matching algorithm for cooperative mapping by autonomous robots. In *Distributed autonomous robotic systems 4*, pages 251–260. Springer, 2000.
7. Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part I. *Robotics & Automation Magazine, IEEE*, 13(2):99–110, 2006.
8. Felix Endres, Jurgen Hess, Jurgen Sturm, Daniel Cremers, and Wolfram Burgard. 3-D mapping with an RGB-D camera. *IEEE Transactions on Robotics*, 30(1):177–187, 2014.
9. Jakob Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam: Large-scale direct monocular slam. In *European Conference on Computer Vision*, pages 834–849. Springer, 2014.
10. M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, June 1981.
11. Dieter Fox, Jonathan Ko, Kurt Konolige, Benson Limketkai, Dirk Schulz, and Benjamin Stewart. Distributed multirobot exploration and mapping. *Proceedings of the IEEE*, 94(7):1325–1339, 2006.
12. Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361, 2012.
13. Marc Pollefeys Gim Hee Lee, Friedrich Fraundorfer. MAV visual SLAM with plane constraint. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 3139–3144, May 2011.
14. Peter Henry, Michael Krainin, Evan Herbst, Xiaofeng Ren, and Dieter Fox. RGB-D mapping: Using kinect-style depth cameras for dense 3D modeling of indoor environments. *International Journal of Robotics Research*, 31(5):647–663, 2012.
15. Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.
16. 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*, 31(2):216–235, 2012.
17. Georg Klein and David Murray. Parallel tracking and mapping for small AR workspaces. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 225–234, 2007.
18. Stefan Kohlbrecher, Oskar Von Stryk, Johannes Meyer, and Uwe Klingauf. A flexible and scalable slam system with full 3d motion estimation. In *Safety, Security, and Rescue Robotics (SSRR), 2011 IEEE International Symposium on*, pages 155–160. IEEE, 2011.
19. Kurt Konolige and Motilal Agrawal. Frameslam: From bundle adjustment to real-time visual mapping. *IEEE Transactions on Robotics*, 24(5):1066–1077, 2008.
20. Rainer Kummerle, Giorgio Grisetti, Hauke Strasdat, Kurt Konolige, and Wolfram Burgard. g 2 o: A general framework for graph optimization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3607–3613, 2011.
21. Yan Lu and Dezhen Song. Visual navigation using heterogeneous landmarks and unsupervised geometric constraints. *IEEE Transactions on Robotics (T-RO)*, 31(3):736–749, June 2015.
22. Lingni Ma, Christian Kerl, Jörg Stückler, and Daniel Cremers. CPA-SLAM: Consistent plane-model alignment for direct RGB-D SLAM. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 1285–1291. IEEE, 2016.
23. Raul Mur-Artal, JMM Montiel, and Juan D Tardós. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.
24. Richard A Newcombe, Dieter Fox, and Steven M Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 343–352, 2015.
25. Paul Newman, David Cole, and Kin Ho. Outdoor SLAM using visual appearance and laser ranging. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 1180–1187. IEEE, 2006.

26. Lina María Paz, Pedro Piniés, Juan D Tardós, and José Neira. Large-scale 6-DOF SLAM with stereo-in-hand. *IEEE Transactions on Robotics*, 24(5):946–957, 2008.
27. Eric Royer, Maxime Lhuillier, Michel Dhome, and Jean-Marc Lavest. Monocular vision for mobile robot localization and autonomous navigation. *International Journal of Computer Vision*, 74(3):237–260, 2007.
28. Renato F Salas-Moreno, Ben Glocker, Paul HJ Kelly, and Andrew J Davison. Dense planar SLAM. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 157–164, 2014.
29. Renato F Salas-Moreno, Richard A Newcombe, Hauke Strasdat, Paul HJ Kelly, and Andrew J Davison. Slam++: Simultaneous localisation and mapping at the level of objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1352–1359, 2013.
30. Hauke Strasdat, JMM Montiel, and Andrew J Davison. Real-time monocular SLAM: Why filter? In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2657–2664, 2010.
31. Hauke Strasdat, JMM Montiel, and Andrew J Davison. Scale drift-aware large scale monocular SLAM. In *Robotics: Science and Systems (RSS)*, volume 1, page 4, 2010.
32. 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)*, pages 5182–5189, 2013.
33. Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. MIT Press, 2005.
34. Ji Zhang and Sanjiv Singh. Loam: Lidar odometry and mapping in real-time. In *Robotics: Science and Systems*, volume 2. Citeseer, 2014.
35. Ji Zhang and Sanjiv Singh. Visual-lidar odometry and mapping: Low-drift, robust, and fast. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 2174–2181. IEEE, 2015.