High Quality Pose Estimation by Aligning Multiple Scans to a Latent Map

Qi-Xing Huang
Stanford University
450 Serra Mall
Stanford, CA 94305

Dragomir Anguelov
Google Inc.
1600 Amphitheatre Pkwy
Mountain View, CA 94043

Abstract— **We introduce a method for fast and accurate registration of multiple horizontal laser scans obtained by a mobile robot. The method is based on novel representation of the scene geometry called a *latent map*, which consists of a set of piecewise linear functions defined over a spatial grid covering the scene. The latent map representation is designed to handle the noise, outliers and limited spatial resolution of laser scan data in a principled manner. The main idea of our algorithm is to iterate between optimizing the latent map, and optimizing the alignment between the latent map and the input scans. In order to handle large datasets containing tens of thousands of scans, we introduce a multi-resolution pose estimation procedure, which applies our scan alignment algorithm at multiple resolutions and combines the results. We demonstrate our novel scan matching technique on several challenging data sets, where we obtain state of the art pose and map reconstructions that are markedly superior to what pairwise scan matching methods can achieve.**

I. INTRODUCTION

Laser range-finders have become standard equipment in mobile robotics and are commonly used for localization and mapping tasks. Laser scan matching can be used not only to estimate the incremental movement of the robot and the environment geometry [10], [16], [29], but also to compute accurate pose constraints when the robot revisits the same scene multiple times [18], [13]. Scan matching is often performed between pairs of scans using variants of the original ICP algorithm [2], [10]. However, this approach can produce errors due to the noise, outliers (caused by moving objects and surface reflectance) and limited angular resolution that are characteristic of laser data. Information from two scans is not sufficient to properly address these issues. Alignment of multiple scans is typically done by adding the scans one at a time to a fixed group of previously added scans. This efficient approach has the drawback that previously computed scan relationships do not get adjusted based on this new data. Alignment of all scans to each other can also be performed by optimizing all overlapping scan pairs, but this approach has quadratic worst-case complexity.

In this paper, we introduce a novel scan matching algorithm, which can efficiently align large groups of scans at the same time. The main idea of the new technique is to maintain an explicit estimate of the scene geometry called a *latent map*, represented as a set of piecewise linear functions (called *surfels*) over a spatial grid at a predefined resolution. Each surfel approximates the surface inside its grid cell using a linear segment. Our algorithm starts by computing a low-resolution version of this latent map from roughly aligned scans (which we expect as input), and then uses the map estimate to compute a better alignment of all the scans. As we continue repeating this process multiple times until convergence, we gradually increase the resolution of the latent map to match the improved scan registration.

Our scan alignment algorithm has several advantages over methods that align pairs or groups of scans to each other and, as a result, suffer from the sparse data and noise present in these scans. Instead, our algorithm explicitly reconstructs a latent map and in the process can utilize data from all scans at the same time. Our latent map combines the strengths of implicit and explicit surface representations, such as fast point correspondence queries and high approximation power. Through its use of piecewise linear segments, our model includes an implicit bias towards straight-line surfaces, which are ubiquitous in man-made environments. Our representation is also able to deal implicitly with outliers such as moving objects, because each surfel in the latent map maintains an estimate of how well it approximates the readings in its corresponding grid cell, and is weighted accordingly when computing the scan alignment.

Our algorithm is very efficient, since it manages to capture a worst-case quadratic number of pairwise scan relationships using a linear number of constraints between each scan and the template surface. This allows us to maintain implicit long-range relationships (for example between scan 1 and scan 100), which minimize pose drift, without incurring quadratic cost. Furthermore, the latent map can be very efficiently encoded in memory using a quadtree data structure. The algorithm generalizes directly to the full 3D case, where the grid is comprised of voxels stored in an octree.

While our scan matching technique is the main focus of the paper, we introduce a specific SLAM algorithm, which is designed to demonstrate the full power of the technique. The algorithm applies our method in a multi-resolution framework that consists of two phases. In the bottom-up phase, groups of scans that are aligned by our algorithm are treated as scans in the next level of the algorithm. In the top-down phase, the alignment results at a higher level are used to generate the initial poses at the current level. In a final step, our algorithm optimizes the alignment of all scans simultaneously. We apply this multi-level SLAM algorithm

to a number of difficult real world datasets, captured both indoors and outdoors and containing vehicles and moving people. We show that our technique is highly efficient and produces globally consistent results for cyclic environments without the need to explicitly close loops.

The remainder of this paper of organized as follows. In Sec. II we review related work. In Sec. III we present our formulation of the multiple scan alignment method. Sec. IV describes how to optimize the formulation's objective. Then in Sec. V we present our multi-resolution pose optimization method. We evaluate our the performance of our method in Sec. VI and we conclude in Sec.VII.

II. Related Work

The main focus of our paper is to present a novel scan matching method, which can be integrated into many different SLAM algorithms. A complete survey of the available SLAM algorithms is outside the scope of this paper, and we refer our readers to [28], [7], [1] for an overview.

Laser scan matching is a popular method of estimating the incremental movement of mobile robots. Typically, scan matching involves aligning the latest acquired scan to a representation of the environment maintained by the robot. The details of this representation vary between the different algorithms. In the simplest case, we can perform matching to the previous acquired scan, or to a group of scans with fixed poses. This is typically done using variants of the ICP method [2], which align scan points to scan points [10] or scan points to line segments [16]. Aligning all scan in a group to each other has been known to produce better results, at the cost of dealing with a worst-case quadratic number of scan pairs [13]. The seminal work of Lu and Milios [18], which is related to our multi-resolution pose estimation framework, maintains a set of pairwise relationships between all overlapping scan pairs, but its worst-case computational complexity is $O(n^3)$ for n scans.

A different set of algorithms uses a representation called occupancy grid maps [8] to aggregate information from multiple scans. Occupancy grid maps are fine-grained metric grids of binary variables that encode whether their cells are free or occupied. It is a compact and simple probabilistic representation that is very efficiently updated with new scans. It is also a great fit for Monte Carlo algorithms, which maintain multiple discrete scan alignment hypotheses using particles to increase the robustness with regards to scan matching errors [26], [9], [12]. (Monte Carlo methods have also been successfully used in conjunction with scan matching [27], but at the cost of increased implementation complexity.) Despite the success of these methods in preventing large alignment errors, the occupancy grid map representation is inherently discrete and does not support the estimation of local gradients, which prevents the robot pose from being determined with high precision. The locally imprecise poses can accumulate to cause large drift over time. As a result, a large number of particles, and diverse sampling schemes ([20], [12]) need to be employed to ensure that the correct trajectory has been sampled.

Another set of approaches represents the environment using a collection of line segments [4]. Long line segments can be helpful to prevent drift in building hallways, however this same bias can hurt in non-planar environments. In addition, estimating the orientation and length of arbitrary segments comes at the cost of increased computational complexity and brittleness. Perhaps for this reason, several methods estimating segment maps (or planar maps) assume that pose has already been obtained using different means [17], [19].

Laser scan registration has been extensively explored in the field of computer graphics for the related problem of recovering 3D surface models from laser scans. Typically in the literature, a step that registers all the input scans [21], [24], [11], [23] is followed by a step that reconstructs the surface using the alignment information [5], [6]. Some algorithms, notably the work of Huang et al. [15] and Ohtake et al. [22], perform both scan alignment and 3D surface reconstruction simultaneously. Our algorithm shares this property, and can be viewed as an application of the ideas of Huang et al. to the robot pose estimation case, with several important differences. Their work was designed to reconstruct watertight 3D surfaces, while we have focused on 3D pose estimation using horizontal lasers. Instead of constructing a 3D latent map from piecewise quadratic functions blended together using B-splines, we use a 2D piecewise-linear representation, which is more efficient and better suited for man-made environments containing lots of straight lines and thin surfaces.

Finally, we introduce a multi-resolution framework, which allows us to handle datasets containing tens of thousands of scans. Our multi-resolution approach is novel in the context of laser-based SLAM. Previous approaches have mostly focused on dealing with the fixed resolution and the growing grid size of occupancy grid maps [25], [30]. These methods incorporate the laser scans sequentially into the map, which results in global drift. In contrast, the global alignment phase of our algorithm deals with global drift by explicitly aligning all laser scans in a given hierarchy level to each other.

III. Probabilistic Framework

Our algorithm is based on a probabilistic framework describing how the laser scans of a moving robot are generated. As illustrated in Fig. 1, we have a robot equipped with a horizontal laser scanner which explores the scene, whose geometry is denoted as M. We will call M the *latent map* of the scene, as it is not provided beforehand to the robot. The robot observations comprise a set of range scans at n poses T_i where $1 \leq i \leq n$. Each pose T_i is a rigid 2D transform relating the robot coordinate system Σ_i to the latent map coordinate system Σ. The scan obtained at pose T_i is denoted as S_i. Each scan S_i contains n_i data points

s_{ij}, where $1 \leq j \leq n_i$. The Cartesian coordinates of each data point s_{ij} in the local coordinate system Σ_i are denoted as $\mathbf{s}_{ij}$. The ray between point $\mathbf{s}_{i,j}$ and the robot location is denoted as as $\mathbf{n}_{ij}$ and provides information about the surface normal at that point.

In this setting, our goal is to estimate the poses $\{T_i\}$ from the input range scans $\{S_i\}$. Our idea is to treat the unknown map M as a hidden variable and optimize it together with the poses $\{T_i\}$. As we will see later, incorporating M into the optimization process enables us the handle the multiple scan alignment problem in a more robust and efficient way.

In probabilistic terms, we formulate multiple scan alignment as the problem of finding the most likely scan poses $\{T_i\}^*$ and latent map M^* that maximize the posterior probability:

$$M^*, \{T_i^*\} = \arg \max_{M,\{T_i\}} P(M, \{T_i\}|\{\mathcal{S}_i\}). \quad (1)$$

Using Bayes' rule, we can expand the posterior probability as follows:

$$P(M, \{T_i\}|\{\mathcal{S}_i\}) = \frac{P(\{\mathcal{S}_i\}|M, \{T_i\}) \cdot P(M, \{T_i\})}{P(\{\mathcal{S}_i\})} \quad (2)$$

Assuming the priors on M and $\{T_i\}$ are independent, and taking the logarithm of the objective gives us:

$$\begin{aligned} M^*, \{T_i^*\} &= \arg \min_{M,\{T_i\}} (-\log P(\{S_i\}|M, \{T_i\}) \\ &- \log P(M) - \log P(\{T_i\})) \end{aligned} \quad (3)$$

The resulting objective function contains three terms: the data likelihood term, the map prior term and the pose prior term. Their definitions are presented in the rest of this section.

A. Data Likelihood Term

To simplify the data likelihood term, we assume that data points are sampled independently from the underlying scene:

$$-\log P(\{S_i\}|\{T_i\}, M) = -\sum_{i=1}^{n} \sum_{j=1}^{n_i} \log P(\mathbf{s}_{ij}|T_i, M) \quad (4)$$

The terms $\log P(\mathbf{s}_{ij}|T_i, M)$ depend on how we parameterize the latent map M. We explore the fact that most scenes are well approximated by a piecewise-linear model and maintain a grid-based representation of the latent map. Each grid cell c_k that contains scan points is associated with a local line segment, called *surfel*, which can be described in terms of a point $\mathbf{p}_k$ and an associated unit normal vector $\mathbf{n}_k$. Then the signed distance from a point $\mathbf{x}$ in the neighborhood of cell c_k to the surfel can be expressed as:

$$d(\mathbf{x}, c_k) = (\mathbf{x} - \mathbf{p}_k)^T \mathbf{n}_k.$$

A standard way to define the data likelihood term is to consider the squared distance from a point to the surfel of its corresponding grid cell. We choose a slightly different formulation, which forces the surfels associated with neighboring cells to be consistent with each other. The set $\mathcal{N}(c_k)$ of each cell c_k is defined as the 3x3 sub-grid centered on c_k. For each point, the term is the sum of squared distances from this point to all its neighboring surfels:

$$-\log P(\mathbf{s}_{ij}|T_i, \mathcal{M}) = \sum_{T_i(\mathbf{s}_{ij}) \in \mathcal{N}(c_k)} \frac{1}{\sigma_k^2} ((T_i(\mathbf{s}_{ij}) - \mathbf{p}_k)^T \mathbf{n}_k)^2. \quad (5)$$

We allow different standard deviations σ_k for different cells, in order to be able to distinguish between objects such as walls, bushes and moving objects. The optimal values of σ_k will be determined through optimization.

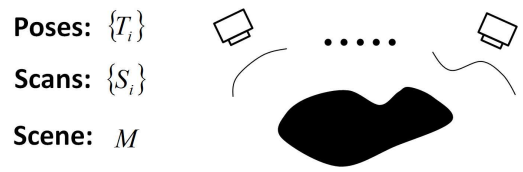


Fig. 1: *Illustration of our formulation to the multiple scan alignment problem.*

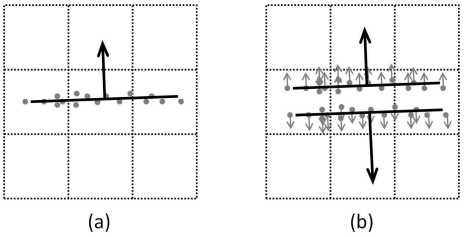


Fig. 2: *Illustration of grid cells containing one and two surfels, respectively.*

Handling of Thin Walls In many indoor cases, there are thin objects such as walls, doors or panels. To prevent points from either side of a thin surface to align to each other would require a very dense grid, which ensures that the two sets of points end up in different cells. In practice, we find that increasing the grid resolution to this level drastically decreases the efficiency of our algorithm. As shown in Fig. 2, our idea is to use two surfels for those cells that contain scan points with opposite normal orientations. In this case, the distance from a data point to each cell is defined as its distance to the surfel, whose normal direction agrees with the point normal.

B. Prior Terms

Map Prior Term If our latent map is an accurate description of the scene, then the standard deviation σ_k of each cell should be close to the standard deviation value σ_0, corresponding to the expected accuracy of the laser scanner. In our experiments we set $\sigma_0 = 0.02$ meters.

Based on this observation, we define our map prior term as

$$-\log P(\mathcal{M}) = \lambda_{\mathcal{M}} \sum_{c_k \in \mathcal{M}} |\sigma_k - \sigma_0|. \quad (6)$$

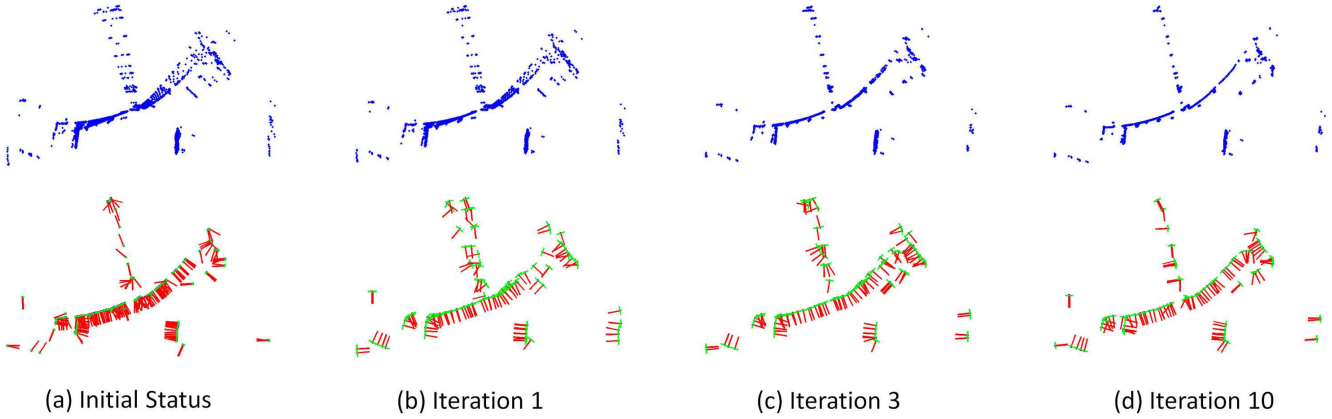


Fig. 3: *Simultaneous alignment of 20 scans using our method. The top row shows the aligned laser scans at different iterations. The bottom row shows the latent maps, where the normal of each surfel is colored in red.*

Here the weight λ_M controls the importance of the map prior. In our experiments, $\lambda_M = 1000$.

Pose Prior Term Wheel encoders are typically available on most robotic platforms. We introduce an optional term that uses the wheel information to constrain the relative vehicle poses. Let $T^w_{i,i+1}$ denote the transformation between poses Σ_i and Σ_{i+1} according to the wheels. Then we have the constraint that $T_{i+1} = T_i \circ T^w_{i,i+1}$. The operator $\circ$ is defined by the following matrix inequalities:

$$\mathbf{R}_{i+1} = \mathbf{R}_i \cdot \mathbf{R}^w_{i,i+1}, \quad \mathbf{t}_{i+1} = \mathbf{R}_i \cdot \mathbf{t}^w_{i,i+1} + \mathbf{t}_i,$$

where $\mathbf{R}$ and $\mathbf{t}$ represent rotation and translation.

The pose prior term that enforces these constraints is:

$$\begin{aligned} -\log P(\{T_i\}) &= \lambda_T \sum_{i=1}^{n-1} (\lambda_R \|\mathbf{R}_{i+1} - \mathbf{R}_i \cdot \mathbf{R}^w_{i,i+1}\|^2_{\mathcal{F}} \\ &+ \|\mathbf{t}_{i+1} - \mathbf{t}_i - \mathbf{R}_i \cdot \mathbf{t}^w_{i,i+1}\|^2_{\mathcal{F}}). \end{aligned} \tag{7}$$

Above, $\|\cdot\|_{\mathcal{F}}$ stands for the Frobenius matrix norm. The parameter λ_T controls the strength of the pose prior term, while λ_R controls the relative importance of the rotation constraints ($\lambda_R = 0.1$ in our experiments).

IV. Optimization

Our strategy is based on alternating between two optimization steps. First, we fix the poses $\{T_i\}$ and optimize the latent map M:

$$M^* = \arg\min_M -\log P(\{S_i\}|M, \{T_i\}) - \log P(M). \tag{8}$$

Then we fix the latent map and optimize the scan poses $\{T_i\}$:

$$\{T_i^*\} = \arg\min_{\{T_i\}} -\log P(\{S_i\}|M, \{T_i\}) - \log P(\{T_i\}). \tag{9}$$

As discussed in [3], the convergence rate of alternating optimization is linear. In our setting, we found that 10-30 iterations is sufficient for convergence. In the rest of this section, we will describe how to perform these two steps.

A. Latent Map Optimization

We combine the terms of our probabilistic model to obtain the following latent map objective:

$$f_{map} = \sum_{c_k} (\sum_{\mathbf{s}'_{ij} \in \mathcal{N}(c_k)} \frac{((\mathbf{s}'_{ij} - \mathbf{p}_k)^T \mathbf{n}_k)^2}{\sigma_k^2} + \lambda_M |\sigma_k - \sigma_0|). \tag{10}$$

In this objective, each cell can be optimized independently. Moreover, the optimal values of $\mathbf{p}$ and $\mathbf{n}$ are independent of σ_k and can be obtained first by solving

$$\mathbf{p}_k^*, \mathbf{n}_k^* = \arg\min_{\mathbf{p},\mathbf{n}} \sum_{\mathbf{s}'_{ij} \in \mathcal{N}(c_k)} ((\mathbf{s}'_{ij} - \mathbf{p}_k)^T \mathbf{n}_k)^2. \tag{11}$$

This equation describes the standard line fitting problem, which can be solved by principal component analysis (PCA) (See Fig. 2(a)). The optimal value of $\mathbf{p}$ is given as the barycenter of the data points:

$$\mathbf{p}_k^* = \sum_{\mathbf{s}'_{ij} \in \mathcal{N}(c_k)} \mathbf{s}'_{ij} / \sum_{\mathbf{s}'_{ij} \in \mathcal{N}(c_k)} 1. \tag{12}$$

The optimal value of $\mathbf{n}$ is found by computing the first eigenvector of the covariance matrix in the following equation:

$$(\sum_{\mathbf{s}'_{ij} \in \mathcal{N}(c_k)} (\mathbf{s}'_{ij} - \mathbf{p}_k^*) \cdot (\mathbf{s}'_{ij} - \mathbf{p}_k^*)^T) \mathbf{n}_k^* = \lambda_k \mathbf{n}_k^* \tag{13}$$

Multiplying both sides of Eq.13 by $\mathbf{n}^{*T}$, we get

$$\lambda_k = \sum_{\mathbf{s}'_{ij} \in \mathcal{N}(c_k)} ((\mathbf{s}'_{ij} - \mathbf{p}_k^*)^T \mathbf{n}_k^*)^2. \tag{14}$$

Thus the optimal value of σ_k is found by solving

$$\sigma_k^* = \arg\min_{\sigma_k} (\frac{\lambda_k}{\sigma_k^2} + \lambda_M |\sigma_k - \sigma_0|). \tag{15}$$

The closed-form solution to the above equation is given by

$$\sigma_k^* = \max(\sigma_0, (\frac{2\lambda_k}{\lambda_M})^{\frac{1}{3}}). \tag{16}$$

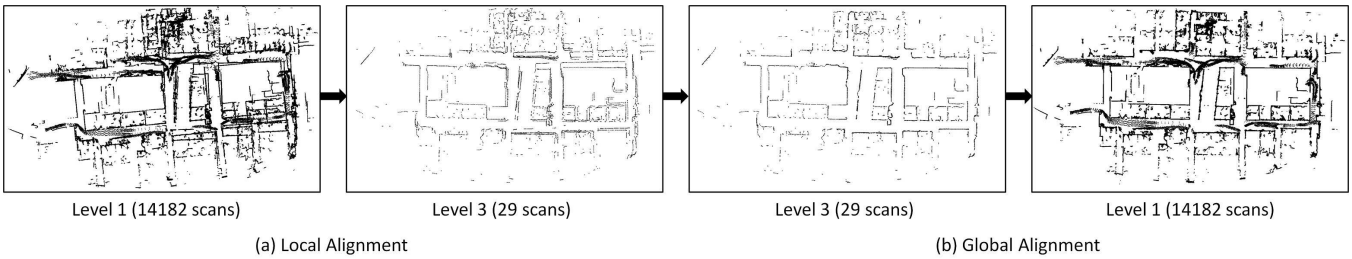


Fig. 4: *Illustration of our multi-resolution pose estimation framework, where we gradually improve the pose estimation quality. (a) In the local alignment phase, we divide scans into overlapping groups and align each group separately. Scans at the next level are formed by the estimated latent maps at the previous level. (b) In the global alignment phase, we align all scans in a level to each other. The alignment results are used to initialize the scan positions at the next level.*

We omit the straightforward derivation of this term.

Eq. 16 provides insight into how our method handles moving objects. Cells containing moving objects will have high λ_k, causing σ_k to be high as well. Then the contribution of these cells to the objective function will be small.

Optimizing Cells Containing Two Surfels. So far we have only considered the case where each cell contains a single surfel. However, the extension to handle cells with two surfels is straightforward. As illustrated in Fig. 2(b), we test whether the data points contained in a given cell can be divided into two groups with opposite normal directions. If this is the case, we compute the surfel parameters for each group separately.

We use a greedy algorithm to split the input points into two groups. We maintain two sets of normal vectors P_1 and P_2 with normals at least 90 degrees apart. Each set P_i is associated with a normal vector $\mathbf{n}_{P_i}$. We incrementally insert each normal vector into one of these two sets. Given a normal $\mathbf{n}$, we insert it into P_1 if $\mathbf{n}_{P_1}^T\mathbf{n} < \mathbf{n}_{P_2}^T\mathbf{n}$ and into P_2 otherwise. Each inserted normal $\mathbf{n}$ is used to update the normal estimate in P_i. We found this greedy procedure works very well in practice.

Grid Resolution For efficiency and stability concerns, we adjust the grid resolution during alignment. We use a coarse grid when we start the optimization to ensure that the neighborhood of each cell is big enough to contain data points from different scans. We slowly increase the grid resolution in subsequent iterations, to capture more scene details when the scans become better aligned.

B. Pose Optimization

Given a fixed latent map M, the pose objective function reduces to the following equation:

$$\begin{aligned} f_{pose} &= \sum_{i=1}^{n}\sum_{j=1}^{n_i}\sum_{\mathbf{s}'_{ij}\in\mathcal{N}(c_k)} \frac{((\mathbf{R}_i\mathbf{s}_{ij}+\mathbf{t}_i-\mathbf{p}_k)^T\mathbf{n}_k)^2}{\sigma_k^2} \\ &+ \lambda_P \sum_{i=1}^{n-1}(\lambda_R\|\mathbf{R}_{i+1}-\mathbf{R}_i\cdot\mathbf{R}^w_{i,i+1}\|^2 \\ &+ \|\mathbf{t}_{i+1}-\mathbf{t}_i-\mathbf{R}_i\cdot\mathbf{t}^w_{i,i+1}\|^2). \end{aligned} \quad (17)$$

This function can be optimized very efficiently using a Gauss-Newton method. We express each 2D rotation matrix $\mathbf{R}_i$ as a function of a counter-clockwise angle θ_i. The first order approximation of $\mathbf{R}_i$ at current angle θ_i^c is given by:

$$\mathbf{R}_i(\theta_i) \approx \mathbf{R}_i(\theta_i^c) - \begin{pmatrix} \sin\theta_i^c & \cos\theta_i^c \\ -\cos\theta_i^c & \sin\theta_i^c \end{pmatrix}(\theta_i-\theta_i^c) \quad (18)$$

Thus, at each step of the Gauss-Newton optimization, we substitute Eq.18 into Eq.17 and optimize f_{pose} to find the optimal values of θ_i and $\mathbf{t}_i$. Note that in this case f_{pose} is a quadratic function and can be optimized by solving a block-diagonal linear system, which takes linear time.

V. Multi-Resolution Framework for Pose Optimization

In this section, we show how to apply our scan alignment method in order to to estimate the robot trajectory and the map of the environment. We focus on the offline pose estimation problem, where we have access to all laser scans collected by the robot. The simplest strategy is to run our scan alignment method directly on all available scans. However, this strategy will most likely fail, because the scan alignment algorithm expects reasonable starting pose estimates. Instead, we leverage the fact that reasonable pose estimates are available locally (for example, by using wheel odometry information) and can be used to bootstrap our algorithm. We improve our pose estimates gradually, by performing scan alignment at several different levels containing progressively larger groups of scans.

Our pose estimation algorithm proceeds in two phases: local alignment phase and global alignment phase. In the local alignment phase, we build the optimization hierarchy. Each level of the hierarchy consists of a set of scans. The scans at the first level are the input scans. To build the scan at the next level, we subdivide the current level scans into overlapping groups and apply our multiple scan alignment algorithm to the scans in each group. The surfels in the latent map solution at the current level are used as scan data for the next level. At each level in the hierarchy, we jointly optimize groups of 20 scans at a time, with the latter levels containing scans of

(a) (b) (c) (d)

Fig. 5: *Our resutls on large indoor datasets. The aligned scans are superposed over the floor plans of the buildings, which were not available to the algorithm at runtime. The zoom-in figures on the bottom row show the details of the latent maps. For each surfel, we display the line segment in green and the normal vector in red. (a) Large office building. (b) Computer History Museum, 1st floor (c) Computer History Museum, 2nd floor (d) Exploratorium Museum, 1st floor.*

much larger size. We stop adding levels when the number of scans becomes less than 20.

In the global alignment phase, we start from the highest level of the hierarchy and use our scan matching algorithm to optimize all the scans contained within that level simultaneously. The alignment result at a higher level is used to generate the initial poses at the previous level. At the final stage of the algorithm, all scans are aligned simultaneously to each other.

An example of our pose estimation is shown in Fig. 4. In this example, we have 14182 laser scans as input, we create 439 groups at the second level and 29 groups at the third level. As can be seen from Fig. 4(a), the relative poses between neighboring scans are improved as we proceed from the first level to the third level. Fig. 4(b) shows that the poses of all scans are made consistent in the global alignment phase.

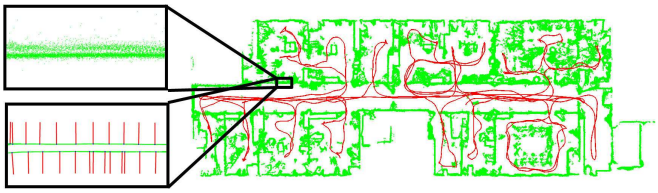


Fig. 6: *Results of our algorithm on example* Albert-b-laser *from the Radish repository [14].*

	scans	points	a_l	a_g	time(s)
Office	14182	2.3M	1.86cm	10cm	412s
CHM 1	41459	7.1M	2.12cm	16cm	612s
CHM 2	35420	6.M	2.34cm	20cm	718s
EXP	60527	9.8M	2.75cm	20cm	1018s
Freiburg	4943	1.8M	2.34cm	NA	200s

TABLE I: *Indoor dataset statistics.*

VI. Experimental Results

Fig. 5 shows the results of our multi-resolution algorithm from Sec. V on several indoor data sets. The datasets were collected using a robot equipped with a horizontal SICK laser scanner running at 30Hz, with an 180-degree field of view and 1 degree angular resolution. In all experimental results, we used the same parameter settings for our algorithm, and obtained high-fidelity maps of the buildings purely by using our multi-resolution method, and without the need to apply loop-closing techniques. For each data set, we evaluated the accuracy of our method in two ways. We computed the average distance from the scan points to the latent map surface, and found that it ranges from 2 cm to 3 cm in these four datasets, which is very close to the laser scanner accuracy. We also used the floorplans of the buildings, which were not available to our algorithm. We examined the alignment between the floorplans and the automatically generated maps, and found that the discrepancy due to global drift was less than 20 centimeters. See Table I for details.

The first dataset, shown in Fig. 5(a), contains many straight walls, a few medium-sized loops and a lot of moving people, which are handled nicely by our method. As can be seen from the top row of Fig. 5(a), the latent map is not well defined in areas with moving people, but we still manage to recover the pose trajectory very well. The second dataset in Fig. 5(b) contains many loops, moving people and a challenging building geometry, but our algorithm manages to provide us with a faithful reconstruction. The third data set contains a large loop, which is successfully closed by our algorithm (see Fig. 5(c)). Finally, Fig. 5(d) shows an example where the shape of the building is mostly curved. Despite our assumption that the surfaces are piecewise linear, we recover an accurate map of this environment.

We also experimented on several indoor sequences from the Radish dataset [14], where we were also able to obtain good results using our original parameter settings. Fig. 6 shows one such result on a dataset from Freiburg with a laser scan rate of 10Hz and an angular resolution of 0.5 degrees.

Although our primary goal is to estimate a globally consistent set of robot poses, the maps generated by our algorithm proved to be accurate even in capturing the fine details of the scene. The bottom row of Fig. 5 collects a set of instances where narrow walls and panels were successfully modeled. This not only underscores the accuracy of our pose estimates, but also the fact that our algorithm is able to correctly prevent the two sides of the narrow surfaces to align to each other. We believe that the line segment and surface normal information present in our maps makes them useful for modeling applications, such as automatic floor plan reconstruction.

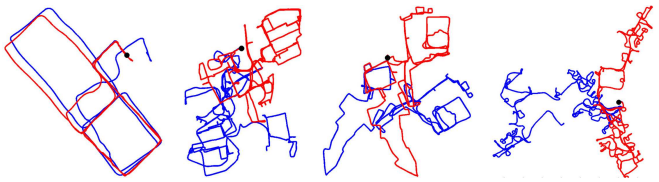
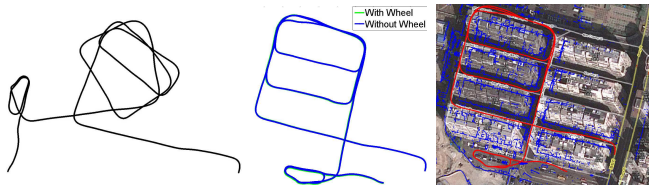


Fig. 7: *Comparison between wheel pose trajectories(blue) and final pose trajectories (red) with their first poses (black) being aligned. Examples are in the same order as in Fig. 5.*

Our method is insensitive to the initial poses. Fig. 7 shows the differences between the initial poses provided by the wheel sensor and the final poses computed by method for the examples shown in Fig.5. It is clear to see that the final poses could be very different from the initial poses.

To further investigate the stability of our method, we tested our algorithm on an outdoor data set where the wheel sensor alone generates fairly poor results (Fig. 8(a)). We tested two sets of motion priors, one provided by the wheels, and one assuming a zero motion prior. Fig. 8(b) shows that our algorithm recovers very similar pose trajectories in both cases, which are also faithful to the map data of that area (Fig. 8(c)).

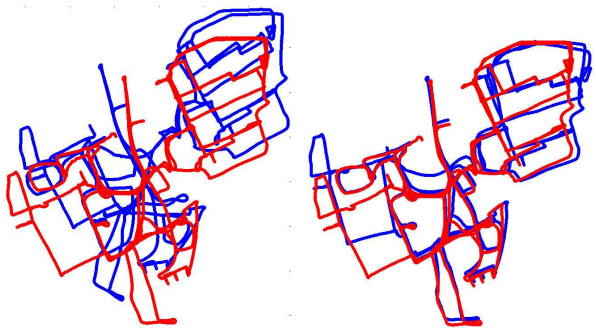


(a) (b) (c)

Fig. 8: *Our algorithm is not very sensitive to the initial poses. (a) The wheel encoder poses have strong drift. (b) The poses after performing scan alignment. Using the wheel prior term and using the static pose prior term produce very similar results. (c) The results agree with the map data.*

Running Time. Our scan alignment method is very efficient. A single run on 20 SICK laser scans takes about 0.03 s. on a computer with a 2.4GHZ CPU. The running time of our multi-scale algorithm takes only about 1/3 of the time spent on acquiring the data sets with the robot. This motivates future work on adapting our multi-scan alignment technique to online pose estimation tasks.

Comparison to Pairwise Matching. We compare our scan matching method to a pairwise ICP implementation that uses the point-to-plane distance metric. For the purpose of this comparison, we ignore our multi-resolution scheme, and simply use our scan matching algorithm on overlapping groups of 10 scans. Fig. 9 shows the drift introduced by the relative poses computed by the two approaches. We can see that our method produces significantly less drift. Furthermore, our algorithm ran 2.5 times faster than its ICP counterpart.



(a) (b)

Fig. 9: *Our scan alignment method produces much less drift than pair-wise scan alignment using ICP. The ground truth is colored red. The estimated pose is colored blue. (a) Pair-wise alignment. (b) Our method.*

Limitations and Future Work Our scan alignment algorithm possesses several limitations. First, it expects reasonable initial pose estimates to be provided. As we demonstrated, our algorithm is robust to small local errors, but it cannot handle the absolute errors that accumulate in large groups of scans. Our multi-resolution framework was designed to address precisely this issue. Second, our algorithm can deal with outliers such as moving objects only if there are enough readings from static objects to counter the forces trying to align the moving object points.

The multi-resolution pose resolution framework we presented is by no means the only algorithm that can be used in conjunction with our general scan alignment framework to solve SLAM problems. It has the known limitation that it cannot close very big loops, when the amount of accumulated drift significantly exceeds the size of the grid cells in our latent map. An area of future work is to experiment with integrating our multi-scan alignment method with several different SLAM frameworks. In particular, we are interested in applying our scan alignment in the context a SLAM algorithm that runs in real time, and in another version that explicitly does loop-closing. Finally, our algorithm can be generalized in a straightforward manner to different laser setups. For example, for robots equipped with rotating lasers such as the Velodyne system, our algorithm can be generalized to obtain a 3D piecewise-planar latent map

representation and 3D pose estimates.

VII. Conclusion

In this paper, we have introduced a novel scan alignment algorithm that can simultaneously align multiple scans to each other. Our novel latent map representation provides a principled way to address the noise, outliers, and low sampling resolution that can be present in laser scan datasets. Its benefits include a well defined distance field, and very efficient ways to correspond data scan points to latent map grid cells. Our method can align a very large number of scans accurately and efficiently, because the complexity of alignment grows only linearly with the number of scans.

We also describe a multi-level SLAM framework based on our scan alignment method. In a number of complex indoor and outdoor environments, our algorithm can recover globally consistent sets of poses and the associated latent maps without the need to perform explicit loop closing. In the process, we demonstrate that our algorithm can handle tens of thousands of scans, while keeping the running time considerably lower than the time it took to acquire those scans in the first place.

Acknowledgements. We thank Dirk Hähnel and Luc Vincent for helpful discussions and Daniel Ratner, Diego Ruspini and Brett Miller for obtaining the experimental datasets.

References

[1] T. Bailey and H. Durrant-Whyte. Simultaneous localization and mapping (SLAM): Part 2. *IEEE Robotics and Automation Magazine*, 13:108–117, 2006.

[2] Paul J. Besl and Neil D. McKay. A method for registration of 3-D shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, 1992.

[3] James C. Bezdek and Richard J. Hathaway. Convergence of alternating optimization. *Neural, Parallel Sci. Comput.*, 11(4):351–368, 2003.

[4] M. Bosse, P. Newman, J. Leonard, and S. Teller. An Atlas framework for scalable mapping. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1899–1906, 2003.

[5] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *SIGGRAPH '96: Proc. of the 23rd annual conference on computer graphics and interactive techniques*, pages 303–312, 1996.

[6] J. Diebel., S. Thrun, and M. Brünig. A Bayesian method for probable surface reconstruction and decimation. *ACM Trans. Graph.*, 25(1):39–59, 2006.

[7] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping (SLAM): Part 1. *IEEE Robotics and Automation Magazine*, 13:99–108, 2006.

[8] A. Elfes. *Occupancy Grids: A Probabilistic Framework for Robot Perception and Navigation*. PhD thesis, Carnegie Mellon University, 1989.

[9] A. Eliazar and R. Parr. DP-SLAM 2.0. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1314– 1320, 2004.

[10] E. Milios F. Lu. Robot pose estimation in unknown environments by matching 2D range scans. *Journal of Intelligent and Robotic Systems*, pages 249–275, 1997.

[11] N. Gelfand, N. Mitra, L. Guibas, and H. Pottmann. Robust global registration. *Symposium on Geometry Processing*, pages 197–206, 2005.

[12] G. Grisetti, C. Stachniss, and W. Burgard. Improving grid-based SLAM with Rao-Blackwellized particle filters by adaptive proposals and selective resamplings. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2432 – 2437, 2005.

[13] J.-S. Gutmann and K. Konolige. Incremental mapping of large cyclic environments. *Proc. of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, pages 318 – 325, 2000.

[14] Andrew Howard and Nicholas Roy. The Robotics Data Set Repository (Radish), 2003.

[15] Q. Huang, B. Adams, and M. Wand. Bayesian surface reconstruction via iterative scan alignment to an optimized prototype. In *Proc. of the fifth Eurographics symposium on Geometry processing*, pages 213–223, 2007.

[16] I.Cox. Blanche an experiment in guidance and navigation of an autonomous robot vehicle. *IEEE Trans. on Robotics and Automation*, 7(2):193–204, 1991.

[17] Y. Liu, R. Emery, D. Chakrabarti, and W. Burgard. Using EM to learn 3D models of indoor environments with mobile robots. In *IEEE International Conference on Machine Learning (ICML*, pages 329–336, 2001.

[18] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4:333–349, 1997.

[19] C. Martin and S Thrun. Online acquisition of compact volumetric maps with mobile robots. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1899–1906, 2002.

[20] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proc. of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1151–1156, Acapulco, Mexico, 2003.

[21] P. J. Neugebauer. Reconstruction of real-world objects via simultaneous registration and robust combination of multiple range images. *International Journal of Shape Modeling*, 3(1-2):71–90, 1997.

[22] Y. Ohtake, A. Belyaev, M. Alexa, and G. Turk. Multi-level partition of unity implicits. *SIGGRAPH*, pages 463–470, 2003.

[23] H. Pottmann, Q. Huang, Y. Yang, and S. Hu. Geometry and convergence analysis of algorithms for registration of 3D shapes. *International Journal of Computation Vision*, 67(3):277–296, 2006.

[24] S. Rusinkiewicz and M. Levoy. Efficient variants of the ICP algorithm. *3DIM*, pages 145–152, 2001.

[25] J. Ryde and H. Hu. 3D mapping with multi-resolution occupied voxel lists. *Autonomous Robots*, 28(2):169–185, 2009.

[26] S. Thrun. A probabilistic online mapping algorithm for teams of mobile robots. *International Journal of Robotics Research*, 20(5):335–363, 2001.

[27] S. Thrun, W. Burgard, and D. Fox. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 321–328, San Francisco, CA, 2000. IEEE.

[28] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. The MIT Press, 2005.

[29] S. Thrun, S. Thayer, W. Whittaker, C. Baker, W. Burgard, D. Ferguson, D. Hähnel, M. Montemerlo, A. Morris, Z. Omohundro, C. Reverte, and W. Whittaker. Autonomous exploration and mapping of abandoned mines. *IEEE Robotics and Automation Magazine*, 11(1):79–91, December 2004.

[30] M. Yguel, O. Aycard, and C. Laugier. Wavelet occupancy grids: a method for compact map building. In *International Conference on Field and Service Robotics*, pages 219–230, 2005.