Fast Registration Based on Noisy Planes with Unknown Correspondences for 3D Mapping

Kaustubh Pathak, *Member, IEEE,* Andreas Birk, *Member, IEEE,* Narūnas Vaškevičius, and Jann Poppinga

***Abstract*—We present a robot pose registration algorithm which is entirely based on large planar surface patches extracted from point-clouds sampled from a 3D sensor. This approach offers an alternative to the traditional point-to-point iterative closest point (ICP) algorithm, its point-to-plane variant, as well as newer grid-based algorithms such as the 3D normal distribution transform (NDT). The simpler case of known plane correspondences is tackled first by deriving expressions for least-squares pose estimation considering plane-parameter uncertainty computed during plane extraction. Closed form expressions for covariances are also derived. To round-off the solution, we present a new algorithm called Minimally Uncertain Maximal Consensus (MUMC) for determining the unknown plane correspondences by maximizing geometric consistency by minimizing the uncertainty volume in configuration space. Experimental results from three 3D sensors, viz. Swiss-ranger, Univ. of South Florida Odetics LADAR, and an actuated SICK S300 are given. The first two have low fields of view (FOV) and moderate ranges, while the third has a much bigger FOV and range. Experimental results show that this approach is not only more robust than point or grid based approaches in plane-rich environments, but it is also faster, requires significantly less memory, and offers a less cluttered planar-patches based visualization.**

I. Introduction

Encouraged by the success of 2D simultaneous localization and mapping (SLAM) [1], there have been many recent attempts (ref. [2], [3], [4], [5], [6]) to extend the methodology to 3D. Since onboard 3D odometry is usually lacking or inadequate, an essential part of the mapping procedure is finding the relative pose offset of the robot between two successive range sensor samples using scan-matching. Typical 3D sensors are laser range finders mounted on a rotating platform [3] and time of flight sensors like the Swiss-ranger [7]. These sensors provide a 3D range scan as a "point-cloud" of noisy spatial coordinates, numbering typically between 10^4 and 10^6.

The most commonly used scan-matching algorithm is the point-to-point (P-P) iterative closest point (ICP) [8], which works with the points directly and hence does not assume any specific structure in the environment. A recent dissertation [9, pp. 4] notes that "*up to now, all approaches successfully applied to 3D SLAM are based on the ICP algorithm*." There also exists a point-to-plane (P-L) version of ICP [10], [11]. Its cost function differs from that of P-P ICP in that it minimizes error along local normals. This increases its robustness to outliers, although no closed-form solution exists for the minimization of its cost function. One alternative to ICP is the 3D normal distribution transform (3D-NDT) [5], which uses a combination of normal distributions defined on an occupancy-grid like structure. This gives a piecewise smooth representation of the point-cloud on which standard iterative numerical optimization methods are applied to obtain the registration.

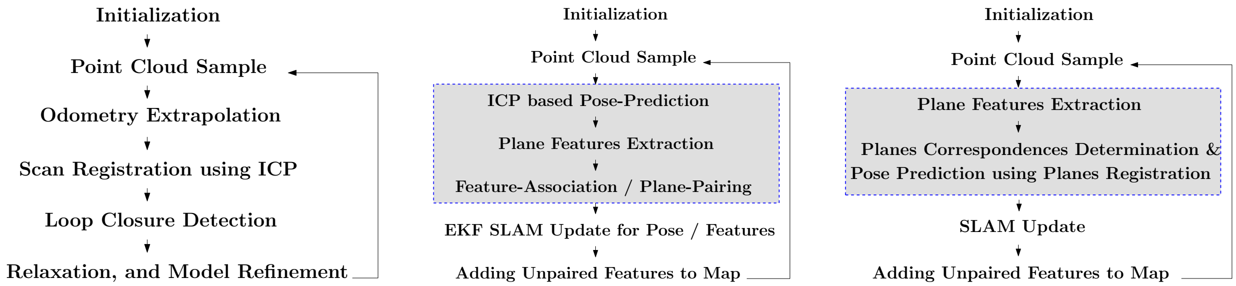
If the environment to be mapped has some structure, e.g, if it is made up of many planar surface patches, then a map based on plane segments offers many advantages in terms of storage requirements, computational efficiency, semantic classification of surface features, and ease of visualization. Furthermore, these features can be easily embedded in an extended Kalman filter EKF-SLAM framework [12]. As in [4], we utilize planar surface-patches. However, our approach obviates the ICP step necessary for the pose change prediction in their work. The basic steps of the algorithms are compared in Fig. 1. Combining plane-correspondence determination and pose registration in one step and removal of ICP leads to savings in computation time and an increase in robustness. In fact, most of the previous 3D mapping approaches have been off-line due to the excessive time required by ICP. With the new approach presented in this article, the pose registration is fast enough relative to the per sample data-acquisition time of the sensor, to be practicable for mapping in a stop-and-go fashion, with relatively large movements of the robot between samples. This contrasts with approaches [13], [14] wherein the sampling is so fast that instantaneous kinematic information may be utilized to aid registration. This also holds for early work on motion determination using camera images utilizing properties of planes in projective geometry [15].

In this article, we focus on the steps marked with a box in Fig. 1 as they are the core elements for 3D mapping. The SLAM step can be, for example, the same as in [4], i.e., the well known EKF algorithm, and is outside the scope of this article. Nevertheless, we have already embedded the approach presented here in a SLAM framework using pose graphs [16]: the resulting 3D maps are presented at http://robotics.jacobs-university.de/projects/3Dmap. For the planes-extraction step and uncertainty analysis, the reader is referred to the authors' previous work [17], [18].

Our approach falls into the category of *large* 3D surface-patches based pose estimation without prior knowledge of correspondences. The work by Kohlhepp et al [2], [19], [20] is most related to ours. In [19], surfaces are extracted from range images obtained by a rotating laser range finder (LRF) and registered together. A local module determines the correspondences and computes transforms, and a global module

This work was supported by the German Research Foundation (DFG).

The authors are with the Dept. of Electrical Engineering and Computer Science, Jacobs University Bremen, 28751 Bremen, Germany. (k.pathak, a.birk)@jacobs-university.de



(a) The approach of Nüchter et al [6]. (b) The approach of Weingarten et al [9]. (c) The approach presented in this article.

Fig. 1. Comparison of algorithm structures. The ICP step in Weingarten et al's approach has been removed in our approach.

detects loop-closure and distributes uncertainty using an elastic graph. For this article, only the local module is relevant which is discussed again in [20]. Similar to Sec. II-C of this article, their approach uses surface-patch directions to compute the rotation between successive views. However, no mention is made in their work of using the uncertainties in the extracted surface-parameters. For estimating translation, they resort back to point features, which is essentially the same as ICP. By contrast, this article uses only planar patches and does not return to the domain of points — even translation is obtained using planes. This allows for detection of dominant directions of uncertainties, as detailed in Sec. II-D. Please note that the uncertainties are absolutely necessary to make the registration usable for proper 3D SLAM. Furthermore, many heuristic measures for estimating correspondences between features, in particular the ground, across views are discussed in [20]. By contrast, the plane correspondence algorithm presented by us does not give any special status to the ground. In our view, their work – although substantial – describes the mathematical machinery insufficiently: it neither computes the uncertainties in feature-parameters in the extracted features, nor discusses how these uncertainties affect the final pose estimation.

A very comprehensive discussion on finding correspondences between two sets of planar or quadratic patches using attribute-graphs is found in [2]. In it, similarity metrics were formulated based on several attributes like shape-factor, area ratio, curvature-histogram, inter-surface relations etc., and a bounded tree search was performed to give a set of correspondences which maximized the metric. The result is refined using an evolutionary algorithm, which is computation-time intensive. In contrast, we use only planar patches and an algorithm is described which maximizes the overall geometric consistency within a search-space to determine correspondences between planes. The search-space is pruned using criteria such as overlap, size-similarity, and agreement with odometry, if available. For all these tests, only the plane parameter covariance matrix is employed, without the need to refer back to the original point-cloud. There is, of course, the option of also using additional attributes like intensity and color for making the correspondence-finding step more reliable. The use of geometric constraints for planar surface matching has been discussed in [21] in which most constraints are formulated as algebraic inequalities *without* employing covariances of plane parameters computed during extraction. Only theoretical and simulation results in a geometric reasoning network were provided without any experimental validation. The general concept of interpretation trees for correspondence finding was introduced in [22], which was later used in [23] for registering planar patches using filters based on area, normal direction, and centroids. The latter work, however, suffers from several shortcomings: 1) their approach is only applicable to *complete* planar patches, i.e. if a patch is only partially visible due to occlusion, it cannot be matched. 2) The actual computation of the registration has not been worked out and it is claimed that only two correspondences are enough for registration. As we show in Sec. II, at least three non-parallel plane correspondences are needed if completeness of patches is not assumed. 3) As experimental evidence, their paper shows the matching of a *single* scan-pair. 4) Again, the covariances of plane parameters were not computed or used. Similarly, the covariance of the registration was not computed. In our work, the covariance of the plane parameters as well as the covariance of the registration solution play a central role in the matching. Another heuristic iterative ICP-like matching algorithm for small planar-patches was developed in [24]. They extract only *small local patches* and apply an ICP like algorithm on these patches– no consensus is built, but several empirical measures are used. Translation is computed using the overlap method similar to [23] which is only valid when there is no occlusion. Correlation in the Fourier domain of *local* surface normals is used in [25] for registration.

Another line of correspondence determination approaches uses Random Sample Consensus (RANSAC) [26], which was employed in [27] in combination with a Huber kernel for matching 2D LIDAR samples. Their algorithm works in 2D and finds the biggest corresponding points set which verifies the inter-point distance rigidity constraint. Since we also use the idea of "consensus" in our approach, it is similar in spirit to RANSAC, albeit with two important differences: there is no random sampling involved, and more importantly, the solution is not solely based on consensus maximization but also uses the uncertainty volume of potential registration solutions.

A. Problem Formulation

In this sub-section, we formulate the problem of plane matching based registration. Two sets of planes are given which are extracted from two successive views of a 3D sensor rigidly mounted on a mobile robot. From these planes, we would like to estimate the change in position and orientation of the sensor between the samples. The problem can be formulated in the following two scenarios with increasing levels of difficulty

1) Both views have the same planes and the correspondences between planes across the views is also given.
2) The two views have only some planes which overlap, i.e., some planes go out of view, and new planes, which were not previously visible, come into view. Which planes are overlapping and what their correspondences are, is not known.

In this article, we address both the cases mentioned above. We start out by deriving a closed form solution for the simplest case of known correspondences in Sec. II. We also derive the covariances for the computed registration. This allows us to compute an uncertainty metric which can be used to measure the geometrical consistency of any assigned correspondences.

The most general case of unknown correspondences is handled in Sec. III, where we propose a new algorithm called the minimally uncertain maximum consensus (MUMC). Some results of applying MUMC to real sensor data are presented in Sec. IV, which also provides a comparison with P-P and P-L ICP, and 3D NDT. Finally, the article is concluded in Sec. V.

II. Planes with Known Correspondences

In this article, the following notation is used:

v, $\mathbf{v}$, $\hat{\mathbf{v}}$	a scalar, a vector, a unit vector.
$\mathbf{M}$, $\mathbf{M}^{+}$	a matrix, its Moore-Penrose pseudo-inverse.
$\lvert\mathbf{M}\rvert$, $\lvert\mathbf{M}\rvert_{+}$	the determinant, the pseudo-determinant.
$\check{\mathbf{q}}$, $\check{\mathbf{q}}^{\star}$	a quaternion, its conjugate.
$\mathbf{u}\cdot\mathbf{v}$, $\check{\mathbf{p}}\diamond\check{\mathbf{q}}$	vector dot product, quaternion product.

The pseudo-determinant of a matrix is simply the product of its non-zero eigenvalues. Additionally, for quantities resolved in different frames, we use the left superscript/subscript notation of [28].

A plane $\mathcal{P}(\hat{\mathbf{m}},\rho)$ is given by the equation $\hat{\mathbf{m}}\cdot\mathbf{p}=\rho$, where ρ is the signed distance from the origin in the direction of the unit plane normal $\hat{\mathbf{m}}$. We see that $\mathcal{P}(\hat{\mathbf{m}},\rho)\equiv\mathcal{P}(-\hat{\mathbf{m}},-\rho)$. To achieve a consistent sign convention, we define planes as $\mathcal{P}(\hat{\mathbf{n}},d)$, where, $d\triangleq|\rho|\geq 0$, and $\hat{\mathbf{n}}\triangleq\sigma(\rho)\,\hat{\mathbf{m}}$, where, $\sigma(\rho)=-1$ if $\rho<0$ and $+1$ otherwise. If $\rho=0$, then we choose the maximum component of $\hat{\mathbf{n}}$ to be positive. The latter case is unlikely to occur in practice in the sensor-frame, because such a plane, which is parallel to the line of sight of the range sensor, is unlikely to be detected by it.

An indexed set ${}^{k}\mathcal{P}$ of planar-patches is extracted [29] from a point-cloud associated with the k-th robot-frame $\mathcal{F}_k$. Apart from the planar patch's $\hat{\mathbf{n}}$ and d parameters, the extraction procedure also gives [17] their 4×4 covariance matrix $\mathbf{C}$. Thus ${}^{k}\mathcal{P}$ is an ordered set of triplets given by

$$ {}^{k}\mathcal{P}\triangleq\{\,{}^{k}\mathcal{P}_i\langle{}^{k}\hat{\mathbf{n}}_i,\,{}^{k}d_i,\,{}^{k}\mathbf{C}_i\rangle,\;i=1\ldots\mathsf{N}_k\}. \tag{1} $$

Additional information, like bounding-boxes or the outlines of the patches may be available, but is not needed in our formulation.

For registration, we consider two robot-frames: a left one denoted as $\mathcal{F}_\ell$ with origin $\mathcal{O}_\ell$ from which the indexed plane-set ${}^{\ell}\mathcal{P}$ is observed, and a right one $\mathcal{F}_r$ with origin $\mathcal{O}_r$ from which the indexed plane-set ${}^{r}\mathcal{P}$ is observed. The equations of the planes are

$$ {}^{\ell}\hat{\mathbf{n}}_i\cdot\,{}^{\ell}\mathbf{p}={}^{\ell}d_i,\qquad\qquad {}^{r}\hat{\mathbf{n}}_i\cdot\,{}^{r}\mathbf{p}={}^{r}d_i. \tag{2} $$

At this juncture, it is assumed that the correspondence problem has already been solved, i.e., the planes at the corresponding index i in the two sets have been found to represent the same physical plane.

If the robot moves from $\mathcal{F}_\ell$ to $\mathcal{F}_r$, and observes the coordinates of the same physical point as ${}^{\ell}\mathbf{p}$ and ${}^{r}\mathbf{p}$ respectively, these coordinates are related by [28]

$$ {}^{\ell}\mathbf{p}={}^{\ell}_{r}\mathbf{R}\,{}^{r}\mathbf{p}+{}^{\ell}_{r}\mathbf{t}, \tag{3} $$

where, the translation ${}^{\ell}_{r}\mathbf{t}\triangleq\overrightarrow{\mathcal{O}_\ell\mathcal{O}_r}$, resolved in $\mathcal{F}_\ell$.

The registration problem now consists of estimating ${}^{\ell}_{r}\mathbf{R}$ and ${}^{\ell}_{r}\mathbf{t}$. Substituting (3) in (2) and comparing coefficients,

$$ {}^{\ell}\hat{\mathbf{n}}_i={}^{\ell}_{r}\mathbf{R}\,{}^{r}\hat{\mathbf{n}}_i, \tag{4a} $$
$$ {}^{\ell}\hat{\mathbf{n}}_i\cdot{}^{\ell}_{r}\mathbf{t}={}^{\ell}d_i-{}^{r}d_i. \tag{4b} $$

We would not have been able to write these equations, had we not enforced the aforementioned sign convention for both sets of planes.

1) Plane Parameter Covariance Matrix: For details on the computation of the plane parameter covariance matrix, the reader is referred to authors' previous work on the extraction of planes from noisy point cloud data; [30] describes the extraction of planes and their polygonal boundaries, [17] deals with the proper calculation of the related uncertainties. The main results are summarized in Appendix A. The 4×4 plane covariance matrix can be partitioned as

$$ \mathbf{C}=\begin{bmatrix}\mathbf{C}_{\hat{\mathbf{n}}\hat{\mathbf{n}}} & \mathbf{C}_{\hat{\mathbf{n}}d}\\ \mathbf{C}_{\hat{\mathbf{n}}d}^{\mathsf{T}} & \mathbf{C}_{dd}\end{bmatrix}. \tag{5} $$

Since $\|\hat{\mathbf{n}}\|=1$, there is uncertainty only in three independent directions. Therefore, this matrix is rank-deficient [31] with rank 3. This, however, does not pose a problem if we interpret results properly. As suggested in [32], we use the Moore-Penrose inverse to obtain the inverse of the covariance matrix (the information matrix). Let the eigenvalue decomposition of the positive semi-definite $\mathbf{C}$ be

$$ \mathbf{C}=\sum_{i=1}^{3}\sigma_i^2\hat{\mathbf{u}}_i\hat{\mathbf{u}}_i^{\mathsf{T}},\;\sigma_1^2\geq\sigma_2^2\geq\sigma_3^2,\;\sigma_4^2=0. \tag{6} $$

We define the following quantities based on the Moore-Penrose inverse:

$$ \mathbf{C}^{+}\triangleq\sum_{i=1}^{3}\frac{1}{\sigma_i^2}\hat{\mathbf{u}}_i\hat{\mathbf{u}}_i^{\mathsf{T}},\qquad\qquad \mathbf{H}\triangleq-\mathbf{C}^{+} \tag{7} $$
$$ |\mathbf{C}|_{+}\triangleq\sigma_1^2\sigma_2^2\sigma_3^2,\qquad\qquad |\mathbf{C}^{+}|_{+}\triangleq|\mathbf{C}|_{+}^{-1} \tag{8} $$

The matrix $\mathbf{H}$ is the Hessian of the log-likelihood function with which the covariance can be estimated, as shown in [33]. In our formulation, the Hessian plays a central role because it, rather than the covariance, is estimated directly in our plane extraction step [30], [17]. Defining,

$$\boldsymbol{\nu} \triangleq \begin{bmatrix}\hat{\mathbf{n}}^\mathsf{T} & d\end{bmatrix}^\mathsf{T}, \tag{9a}$$

$${}^{r}_{\ell}\mathbf{T} \triangleq \begin{bmatrix} {}^{\ell}_{r}\mathbf{R}^\mathsf{T} & \mathbf{0}_{3\times 1} \\ -{}^{\ell}_{r}\mathbf{t}^\mathsf{T} & 1 \end{bmatrix}, \text{ we get,} \qquad {}^{r}\boldsymbol{\nu} = {}^{r}_{\ell}\mathbf{T}\,{}^{\ell}\boldsymbol{\nu}. \tag{9b}$$

The maximum-likelihood registration is then found by doing the following extremization

$$\min_{{}^{r}_{\ell}\mathbf{T}} \zeta_T \triangleq \frac{1}{2}\sum_{i=1}^{\mathsf{N}} ({}^{r}\boldsymbol{\nu}_i - {}^{r}_{\ell}\mathbf{T}\,{}^{\ell}\boldsymbol{\nu}_i)^\mathsf{T}\mathbf{C}_i^{+}({}^{r}\boldsymbol{\nu}_i - {}^{r}_{\ell}\mathbf{T}\,{}^{\ell}\boldsymbol{\nu}_i) + \frac{1}{2}\sum_{i=1}^{\mathsf{N}} \log|\mathbf{C}_i|_+, \text{ where,} \tag{10a}$$

$$\mathbf{C}_i \triangleq {}^{r}\mathbf{C}_i + {}^{r}_{\ell}\mathbf{T}\,{}^{\ell}\mathbf{C}_i\,{}^{r}_{\ell}\mathbf{T}^\mathsf{T} \tag{10b}$$

Due to the coupling of rotation and translation estimation, this is hard to tackle analytically. Iterative numerical optimization could be employed – however, in this article we are looking for closed-form solutions which can be employed for online planes-based mapping. Therefore, we look for fast-to-compute alternatives using decoupling of covariances and application of least-squares.

A. Plane Parametrization Choice

In [9, pp. 29] several plane parametrizations are listed. Only the $\hat{\mathbf{n}}, d$ parameterization used in this article is free of singularities and can represent all possible planes. The 4×4 covariance matrix of this representation is, however, singular. This does not actually represent any difficulty and singular covariance matrices are used routinely in classic works in Computer Vision like [32]. In our approach [17], we go one step further and *exploit* this singularity to place the plane's parameters exactly in the null space of its covariance.

In [9] an inefficient work-around is used. They first use the $\hat{\mathbf{n}}, d$ plane parameterization for plane-fitting using principal component analysis, then rotate all the points such that the plane's normal becomes the $\hat{\mathbf{z}}$ axis, and then refit the plane using a different model $z = ax + by + d$ using least-squares to compute a 3×3 covariance matrix for a, b, d. This is particularly wasteful because of the plane-fitting problem being effectively solved twice. Although the 3×3 covariance matrix is non-singular, the parameterization on which it is based is local to the plane and cannot represent all planes. Hence we are of the opinion that the singularity of the covariance matrix should not be avoided because it makes explicit the underlying topology of the rotation group $SO(2)$ (or $SO(3)$ later for the unit-quaternions).

B. Decoupling the Covariances

To be able to use the nicely decoupled equations (4) for determining rotation and translation separately, we need to estimate the total uncertainty in $\hat{\mathbf{n}}$ by marginalizing, i.e. integrating out the effect of d and vice-versa. The details of this decoupling are shown in Appendix B. This gives us two decoupled covariances: A scalar variance $\mathbf{D}_{dd}$ for d, and a 3×3 covariance $\mathbf{D}_{\hat{\mathbf{n}}\hat{\mathbf{n}}}$ for $\hat{\mathbf{n}}$.

Armed with the above expressions for decoupled covariances, we can go back to the decoupled Eqs. (4) to find the registration. Henceforth in this article, we will mostly use these *decoupled* covariances denoted by $\mathbf{D}$ as given in (58) and (61) for the plane parameter covariances. When the original *coupled* covariances are needed, e.g. in Sec. III-A1, and III-A4, we will use the symbol $\mathbf{C}$.

C. Least Squares Rotation After Decoupling and Assuming Isotropic Uncertainties

In this section, we derive a solution for computing ${}^{\ell}_{r}\mathbf{R}$. Considering first the uncertainty for any given normal to be directionally uniform/isotropic, but varying only in its magnitude leads to a log-likelihood function which is straightforward to analyze. We define the rotational residual for the i-th correspondence as

$$\mathbf{s}_i \triangleq {}^{\ell}\hat{\mathbf{n}}_i - {}^{\ell}_{r}\mathbf{R}\,{}^{r}\hat{\mathbf{n}}_i. \tag{11}$$

Due to the isotropy approximation, it is assumed to be normally distributed with mean $\mathbf{0}$ and covariance $w_i^{-1}\mathbf{I}_3$, where w_i are weights inversely proportional to a measure of rotational uncertainty. Using the property that trace is unitary invariant, the weights are selected as

$$w_i^{-1} = \text{trace}({}^{\ell}\mathbf{D}_{i,\hat{\mathbf{n}}\hat{\mathbf{n}}} + {}^{\ell}_{r}\mathbf{R}\,{}^{r}\mathbf{D}_{i,\hat{\mathbf{n}}\hat{\mathbf{n}}}\,{}^{\ell}_{r}\mathbf{R}^\mathsf{T}) \tag{12}$$

$$= \text{trace}({}^{\ell}\mathbf{D}_{i,\hat{\mathbf{n}}\hat{\mathbf{n}}} + {}^{r}\mathbf{D}_{i,\hat{\mathbf{n}}\hat{\mathbf{n}}}). \tag{13}$$

We therefore need to maximize the log-likelihood function

$$\max_{{}^{\ell}_{r}\mathbf{R}} \zeta_r = -\frac{1}{2}\sum_{i=1}^{\mathsf{N}} w_i \|\mathbf{s}_i\|^2 \tag{14}$$

$$\equiv \text{const.} + \sum_{i=1}^{\mathsf{N}} w_i\,{}^{\ell}\hat{\mathbf{n}}_i \cdot ({}^{\ell}_{r}\mathbf{R}\,{}^{r}\hat{\mathbf{n}}_i). \tag{15}$$

There are essentially two ways to solve this – by using quaternions [34, Sec. 4], and by using SVD [32]. In the context of satellite attitude estimation using a star-tracker, this problem is called the Wahba's problem [35] and the quaternion based solution is called the Davenport's q-method.

We employ the quaternion based approach here as it leads to an easier formulation of the anisotropic uncertainty version in the next section. The solution involves representing the rotation operator ${}^{\ell}_{r}\mathbf{R}$ using a unit-quaternion ${}^{\ell}_{r}\check{\mathbf{q}}$. The unit normal vector $\hat{\mathbf{n}}$ now becomes the purely imaginary quaternion $\check{\mathbf{n}} = \begin{bmatrix}0 & \hat{\mathbf{n}}^\mathsf{T}\end{bmatrix}^\mathsf{T}$. Then (4a) can be rewritten as ${}^{\ell}\check{\mathbf{n}}_i = {}^{\ell}_{r}\check{\mathbf{q}} \diamond {}^{r}\check{\mathbf{n}}_i \diamond {}^{\ell}_{r}\check{\mathbf{q}}^\star$.

After dropping the constant, Eq. (15) can be written as

$$\begin{aligned}\max_{{}^{\ell}_{r}\check{\mathbf{q}}} \zeta_r &= \sum_{i=1}^{\mathsf{N}} w_i({}^{\ell}_{r}\check{\mathbf{q}} \diamond {}^{r}\check{\mathbf{n}}_i \diamond {}^{\ell}_{r}\check{\mathbf{q}}^\star) \cdot {}^{\ell}\check{\mathbf{n}}_i \\ &= {}^{\ell}_{r}\check{\mathbf{q}}^\mathsf{T}\left(\sum_{i=1}^{\mathsf{N}} w_i \overline{\boldsymbol{\Psi}}^\mathsf{T}({}^{r}\check{\mathbf{n}}_i)\boldsymbol{\Psi}({}^{\ell}\check{\mathbf{n}}_i)\right){}^{\ell}_{r}\check{\mathbf{q}}, \\ &\triangleq {}^{\ell}_{r}\check{\mathbf{q}}^\mathsf{T}\mathbf{K}\,{}^{\ell}_{r}\check{\mathbf{q}},\end{aligned} \tag{16}$$

where, the following definitions have been used. Let the quaternion $\check{\mathbf{p}} \triangleq \begin{bmatrix} p_0 & p_x & p_y & p_z \end{bmatrix}^\mathsf{T} \triangleq \begin{bmatrix} p_0 & \mathbf{p}^\mathsf{T} \end{bmatrix}^\mathsf{T}$, then,

$$\check{\mathbf{p}} \diamond \check{\mathbf{q}} \triangleq \mathbf{\Psi}(\check{\mathbf{p}})\check{\mathbf{q}} \triangleq \begin{bmatrix} p_0 & -\mathbf{p}^\mathsf{T} \\ \mathbf{p} & p_0\mathbf{I}_3 + \mathbf{p} \times \mathbf{I}_3 \end{bmatrix} \check{\mathbf{q}}, \tag{17a}$$

$$\check{\mathbf{q}} \diamond \check{\mathbf{p}} \triangleq \overline{\mathbf{\Psi}}(\check{\mathbf{p}})\check{\mathbf{q}} \triangleq \begin{bmatrix} p_0 & -\mathbf{p}^\mathsf{T} \\ \mathbf{p} & p_0\mathbf{I}_3 - \mathbf{p} \times \mathbf{I}_3 \end{bmatrix} \check{\mathbf{q}}. \tag{17b}$$

$\mathbf{p} \times \mathbf{I}_3$ is the cross-product skew-symmetric matrix [32]. The maximum of ζ_r is then achieved at the 4×1 unit Eigenvector ${}^\ell_r\check{\mathbf{q}}_{\mathrm{LS}}$ of $\mathbf{K}$ corresponding to the maximum positive Eigenvalue of the 4×4 symmetric matrix $\mathbf{K}$.

From this computed ${}^\ell_r\check{\mathbf{q}}_{\mathrm{LS}}$, one can get the rotation matrix ${}^\ell_r\mathbf{R}_{\mathrm{LS}} = \mathcal{R}({}^\ell_r\check{\mathbf{q}}_{\mathrm{LS}})$, where, the operator $\mathcal{R}(\check{\mathbf{q}} = [q_0, \mathbf{q}^\mathsf{T}]^\mathsf{T})$ is defined as

$$\mathcal{R}(\check{\mathbf{q}}) \triangleq (q_0^2 - \|\mathbf{q}\|^2)\mathbf{I}_3 + 2\mathbf{q}\mathbf{q}^\mathsf{T} + 2q_0\mathbf{q} \times \mathbf{I}_3 \tag{18}$$

1) Rotational Covariance Estimation: Judging from the recent literature in space and aircraft-dynamics community [36], [37], many ways of representing uncertainty in unit quaternions exist. Since the rotation covariance was not estimated as part of points registration [34], [8], we present a new solution which is based on the Hessian of the cost function [33]. As was the case with the covariance matrix of plane-parameters, the covariance ${}^\ell_r\mathbf{C}_{\check{\mathbf{q}}\check{\mathbf{q}}}$ of the unit quaternion ${}^\ell_r\check{\mathbf{q}}_{\mathrm{LS}}$ has the feature that the matrix is singular [36], [37] due to the unit-norm constraint. In this article, we will use this singular 4×4 matrix, which represents the uncertainty of a unit quaternion on the tangent plane of a unit 3-sphere in 4D.

As there is no uncertainty along the direction of the unit quaternion, ${}^\ell_r\mathbf{C}_{\check{\mathbf{q}}\check{\mathbf{q}}}\,{}^\ell_r\check{\mathbf{q}} = 0$. Such a covariance matrix can be found by computing the Hessian of the constrained Lagrangian $\mathcal{L} \triangleq \zeta_r - \lambda(\check{\mathbf{q}}^\mathsf{T}\check{\mathbf{q}} - 1)$ at $\check{\mathbf{q}} = {}^\ell_r\check{\mathbf{q}}_{\mathrm{LS}}$ and $\lambda_{\mathrm{LS}} = \mu_{\max}(\mathbf{K})$, the latter being the maximum Eigenvalue of $\mathbf{K}$. This Hessian is given by

$$\mathbf{H}_{\check{\mathbf{q}}\check{\mathbf{q}}}({}^\ell_r\check{\mathbf{q}}_{\mathrm{LS}}) \triangleq 2\left(\mathbf{K} - \mu_{\max}(\mathbf{K})\mathbf{I}_4\right). \tag{19}$$

The unit quaternion ${}^\ell_r\check{\mathbf{q}}_{\mathrm{LS}}$ spans the null-space of this Hessian. Finally, the sought covariance is

$${}^\ell_r\mathbf{C}_{\check{\mathbf{q}}\check{\mathbf{q}}} = -\mathbf{H}^+_{\check{\mathbf{q}}\check{\mathbf{q}}}({}^\ell_r\check{\mathbf{q}}_{\mathrm{LS}}). \tag{20}$$

An alternative method for obtaining the covariance is that described in [19] which uses results on perturbation of Eigenvectors.

D. Least Squares Translation After Decoupling

Stacking (4b) for $i = 1 \dots \mathsf{N}$, the translation ${}^\ell_r\mathbf{t}$ should ideally satisfy

$$\mathbf{M}\,{}^\ell_r\mathbf{t} = \mathbf{d}, \tag{21}$$

$$\mathbf{M}_{\mathsf{N}\times 3} \triangleq \begin{bmatrix} {}^\ell\hat{\mathbf{n}}_1^\mathsf{T} \\ \vdots \\ {}^\ell\hat{\mathbf{n}}_\mathsf{N}^\mathsf{T} \end{bmatrix}, \qquad \mathbf{d}_{\mathsf{N}\times 1} \triangleq \begin{bmatrix} {}^\ell d_1 - {}^r d_1 \\ \vdots \\ {}^\ell d_\mathsf{N} - {}^r d_\mathsf{N} \end{bmatrix}$$

Since there is uncertainty in $\mathbf{M}$ as well as $\mathbf{d}$, ideally this should be solved using the method of total least squares (TLS) [38]. However, the standard TLS formulation cannot be used directly because 1) the uncertainty in $\mathbf{M}$ is row-wise correlated and anisotropic and 2) the matrix $\mathbf{M}$ consists of unit vectors and therefore the uncertainty is structured. To our knowledge, the latter problem has not been addressed in the literature in the context of TLS.

Due to its intuitive nature and fast closed-form solution, we will solve Eq. (21) with ordinary least squares (LS). This assumes that all of the uncertainty is on the right hand side, that is in $\mathbf{d}$. A justification for this is provided by Eq. (58), which shows that the uncertainties on the two sides of (21) are related.

The diagonal weighting matrix $\mathbf{W}$ is defined as

$$\mathbf{C}_\mathbf{d} \triangleq \begin{bmatrix} {}^\ell\mathbf{D}_{1,dd} + {}^r\mathbf{D}_{1,dd} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & {}^\ell\mathbf{D}_{\mathsf{N},dd} + {}^r\mathbf{D}_{\mathsf{N},dd} \end{bmatrix}, \tag{22}$$

$$\mathbf{W} \triangleq \left(\mathbf{C}_\mathbf{d}^{-1}\right)^{1/2}. \tag{23}$$

Then the LS solution minimizes $\|\mathbf{W}(\mathbf{M}\mathbf{t}_{r/\ell} - \mathbf{d})\|$. If $\mathbf{M}$ is full rank, the least-squares optimum translation is ${}^\ell_r\mathbf{t}_{\mathrm{LS}} = \left(\mathbf{M}^\mathsf{T}\mathbf{W}^2\mathbf{M}\right)^{-1}\mathbf{M}^\mathsf{T}\mathbf{W}^2\mathbf{d}$.

Unlike rotation, in general we need $\mathsf{N} \geq 3$ mutually non-parallel planes to find ${}^\ell_r\mathbf{t}_{\mathrm{LS}}$. The least-squares formula is not a good way to compute the solution because $\mathbf{M}$ may be ill-conditioned, may be rank-deficient, or $\mathsf{N} < 3$. A more general way to solve the equation is presented next. We define

$$\hat{\mathbf{M}} \triangleq \mathbf{W}\mathbf{M}, \qquad \hat{\mathbf{d}} \triangleq \mathbf{W}\mathbf{d}. \tag{24}$$

Let the singular-value decomposition of $\hat{\mathbf{M}}$ be given by $\mathbf{U}_{\mathsf{N}\times\mathsf{N}}\mathbf{\Lambda}_{\mathsf{N}\times 3}\mathbf{V}^\mathsf{T}_{3\times 3}$. $\mathbf{\Lambda}$ has non-negative singular values σ_i^2 arranged in descending order. The column unit vectors of $\mathbf{U}$ are denoted $\mathbf{u}_i,\ i = 1 \dots \mathsf{N}$ and the column unit vectors of $\mathbf{V}$ are denoted $\mathbf{v}_i,\ i = 1 \dots 3$.

Let $\mathsf{N}_r \leq 3$ be the *effective rank* of $\hat{\mathbf{M}}$. If the largest singular value $\sigma_1^2 < \epsilon_1$, then the effective rank is 0. The parameter ϵ_1 is dependent on machine accuracy. If $\sigma_1^2 \geq \epsilon_1$, then the effective rank is found by finding the count of all singular values $\sigma_i^2 > \sigma_1^2/\bar{c}$ in the diagonal matrix $\mathbf{\Lambda}$, where $\bar{c}$ is the maximum allowable condition number of the matrix. In practice ϵ_1 and $\bar{c}$ are quite important parameters for obtaining good translation estimates and also for identifying the directions in which the translation estimate is the most uncertain.

Then the best rank N_r approximation of $\hat{\mathbf{M}}$ is

$$\tilde{\mathbf{M}} = \sum_{i=1}^{\mathsf{N}_r} \sigma_i^2 \hat{\mathbf{u}}_i \hat{\mathbf{v}}_i^\mathsf{T}, \qquad \mathsf{N}_r \leq 3. \tag{25}$$

The span of the orthogonal unit vectors $\hat{\mathbf{u}}_i,\ i = 1 \dots \mathsf{N}_r$ gives the best approximation for the range-space of $\hat{\mathbf{M}}$. Therefore, the closest we can get to $\hat{\mathbf{d}}$ is $\tilde{\mathbf{d}} = \sum_{i=1}^{\mathsf{N}_r}(\hat{\mathbf{u}}_i \cdot \hat{\mathbf{d}})\hat{\mathbf{u}}_i$, which gives the corresponding translation estimate

$${}^\ell_r\mathbf{t}_{\mathrm{LS}} = \sum_{i=1}^{\mathsf{N}_r} \frac{(\hat{\mathbf{u}}_i \cdot \hat{\mathbf{d}})\hat{\mathbf{v}}_i}{\sigma_i^2} \triangleq \hat{\mathbf{M}}^+\hat{\mathbf{d}}, \quad \hat{\mathbf{M}}^+ \triangleq \sum_{i=1}^{\mathsf{N}_r} \frac{\hat{\mathbf{v}}_i\hat{\mathbf{u}}_i^\mathsf{T}}{\sigma_i^2}. \tag{26}$$

This is also the minimum 2-norm solution of the LS problem regardless of the rank of $\mathbf{M}$ mentioned in [38].

Note that for directions $\hat{\mathbf{v}}_i,\ i = \mathsf{N}_r + 1 \dots 3$, we have no information about the translation. One option is to keep these components 0 and inject large uncertainty along those directions in the covariance matrix. However, if an estimate by overlap (ref. Sec. II-E) or by odometry ${}^{\ell}_{r}\mathbf{t}_e$, along with its covariance matrix $\mathbf{C}_{\mathbf{t},e}$ is available, we can use it *only* for these missing components. In this case, we have

$$\begin{aligned} {}^{\ell}_{r}\mathbf{t}_{\text{LS}} &= \hat{\mathbf{M}}^{+}\hat{\mathbf{d}} + \sum_{i=\mathsf{N}_r+1}^{3} ({}^{\ell}_{r}\mathbf{t}_e \cdot \hat{\mathbf{v}}_i)\,\hat{\mathbf{v}}_i, \\ &\triangleq \hat{\mathbf{M}}^{+}\hat{\mathbf{d}} + \mathbf{M}_e\,{}^{\ell}_{r}\mathbf{t}_e, \quad \mathbf{M}_e \triangleq \sum_{i=\mathsf{N}_r+1}^{3} \hat{\mathbf{v}}_i\hat{\mathbf{v}}_i^{\mathsf{T}}. \end{aligned} \tag{27}$$

Finally, we can write the estimate of the covariance matrix for translation as follows

$$\begin{aligned} {}^{\ell}_{r}\mathbf{C}_{\mathbf{tt}} &= \tau\,\hat{\mathbf{M}}^{+}\mathbf{W}\mathbf{C}_{\mathbf{d}}\mathbf{W}^{\mathsf{T}}(\hat{\mathbf{M}}^{+})^{\mathsf{T}} + \mathbf{M}_e\mathbf{C}_{\mathbf{t},e}\mathbf{M}_e^{\mathsf{T}} \\ &= \tau\,\mathbf{M}^{+}\mathbf{C}_{\mathbf{d}}(\mathbf{M}^{+})^{\mathsf{T}} + \mathbf{M}_e\mathbf{C}_{\mathbf{t},e}\mathbf{M}_e^{\mathsf{T}}, \end{aligned} \tag{28}$$

where, the last equation comes from simplification using (23) and (24), and

$$\tau \triangleq \frac{1}{\mathsf{N} - \mathsf{N}_r}\|\hat{\mathbf{M}}\,{}^{\ell}_{r}\mathbf{t}_{\text{LS}} - \hat{\mathbf{d}}\|^2 \tag{29}$$

In this work, we have used the overlap estimate described next in Sec. II-E to supply the translation along missing $3-\mathsf{N}_r$ directions instead of the odometry since in general, overlap estimates tend to be much more accurate than odometry.

E. Roughly Estimating Translation by Overlap

The previous Sec. II-D provided an accurate method of determining translation when planar patches with normals in all directions are present. When, however, corresponding planes normal to certain directions are missing, we need to estimate the translation along these directions using other methods which are less accurate. In this section, we consider one such method based on the assumption of complete overlap of planes. In general, this will only be accurate if there are no occlusions present and sensor sampling is uniform. Additionally, this method requires that a rotation estimate of ${}^{\ell}_{r}\mathbf{R}$ be known.

Fortunately, as shown in Appendix A, all the required information for using overlap is already present in the 4×4 plane parameter covariance matrix $\mathbf{C} = -\mathbf{H}^{+}$. Letting $\mathbf{H}_{dd}$ denote the last diagonal element of $\mathbf{H}$, the weighted center of mass $\mathbf{p}_c$ of the patch can be computed by (50b), and the scatter of the points *on the patch-plane* (not normal to it) is given by $(-1/\mathbf{H}_{dd})(\mathbf{D}_{\hat{\mathbf{n}}\hat{\mathbf{n}}})^{+}$. This can be seen by using Eqs. (50) and (49) in the definition Eq. (61), which gives

$$\begin{aligned} (-1/\mathbf{H}_{dd})(\mathbf{D}_{\hat{\mathbf{n}}\hat{\mathbf{n}}})^{+} &= \frac{1}{\sum_j w_j}\sum_j w_j(\mathbf{p}_j - \mathbf{p}_c)(\mathbf{p}_j - \mathbf{p}_c)^{\mathsf{T}} \\ &- \frac{1}{\sum_j w_j}\sum_j w_j\hat{\mathbf{n}}^{\mathsf{T}}(\mathbf{p}_j - \mathbf{p}_c)(\mathbf{p}_j - \mathbf{p}_c)^{\mathsf{T}}\hat{\mathbf{n}}\,\mathbf{I}, \end{aligned} \tag{30}$$

where, the summation is on all points belonging to the planar patch with normal $\hat{\mathbf{n}}$. To this we can add the uncertainty along the direction of the normal and define the total overlap uncertainty as

$$\mathbf{C}_{\mathbf{pp}} \triangleq (-1/\mathbf{H}_{dd})(\mathbf{D}_{\hat{\mathbf{n}}\hat{\mathbf{n}}})^{+} + \mathbf{D}_{dd}(\hat{\mathbf{n}}\hat{\mathbf{n}}^{\mathsf{T}}). \tag{31}$$

Note that we take the scatter as a conservative measure of uncertainty because the whole patch may not be visible in the sample owing to occlusion. This is usually the case in practice.

Suppose we are given a set of pairs of corresponding planes, with point-cloud centers back-computed by (50b) as ${}^{\ell}\mathbf{p}_{c,i}$, ${}^{r}\mathbf{p}_{c,i}$, $i = 1\ldots\mathsf{N}$, and suppose further that the rotation ${}^{\ell}_{r}\mathbf{R}$ is known beforehand. The translation and its covariance for the i-th pair can then be estimated as

$${}^{\ell}_{r}\mathbf{t}_i \approx {}^{\ell}\mathbf{p}_{c,i} - {}^{\ell}_{r}\mathbf{R}\,{}^{r}\mathbf{p}_{c,i}, \tag{32}$$

$${}^{\ell}_{r}\mathbf{C}_{\mathbf{t},i} \approx {}^{\ell}\mathbf{C}_{\mathbf{pp},i} + {}^{\ell}_{r}\mathbf{R}\,{}^{r}\mathbf{C}_{\mathbf{pp},i}\,{}^{\ell}_{r}\mathbf{R}^{\mathsf{T}}. \tag{33}$$

The bigger the plane-patch, the higher is the variance in the translation estimate found using it. We have N such estimates. We can then estimate the translation from overlap as

$$\begin{aligned} {}^{\ell}_{r}\mathbf{t}_e &= \arg\min_{\mathbf{t}} \chi^2_e(\mathbf{t}), \\ \chi^2_e(\mathbf{t}) &\triangleq \frac{1}{2}\sum_i^{\mathsf{N}}({}^{\ell}_{r}\mathbf{t}_i - \mathbf{t})^{\mathsf{T}}({}^{\ell}_{r}\mathbf{C}_{\mathbf{t},i})^{-1}({}^{\ell}_{r}\mathbf{t}_i - \mathbf{t}). \end{aligned} \tag{34}$$

The standard closed form solution for this is

$$\mathbf{A} \triangleq \sum_{i=1}^{\mathsf{N}}({}^{\ell}_{r}\mathbf{C}_{\mathbf{t},i})^{-1}, \qquad \mathbf{b} \triangleq \sum_{i=1}^{\mathsf{N}}({}^{\ell}_{r}\mathbf{C}_{\mathbf{t},i})^{-1}\,{}^{\ell}_{r}\mathbf{t}_i. \tag{35}$$

$${}^{\ell}_{r}\mathbf{t}_e = \mathbf{A}^{-1}\mathbf{b}. \tag{36}$$

The optimistic covariance of this estimate is

$$\mathbf{C}_{\mathbf{t},e} = \frac{\kappa\,\chi^2_e({}^{\ell}_{r}\mathbf{t}_e)}{3\mathsf{N}-3}\left(\sum_{i=1}^{\mathsf{N}}({}^{\ell}_{r}\mathbf{C}_{\mathbf{t},i})^{-1}\right)^{-1}, \tag{37}$$

where, $\kappa > 1$ accounts for the presence of occlusions in the scene.

III. Assigning Unknown Correspondences

Up until now we have assumed that the correspondences between the planes in the two successive views are known. Now we drop this assumption and explore ways of answering "which plane is which?" If the ith plane in the left frame corresponds to the jth one in the right, it will be denoted as ${}^{\ell}\mathcal{P}_i \leftrightarrow {}^{r}\mathcal{P}_j$, abbreviated $i \leftrightarrow j$. We first consider some simple tests which help us decide whether $i \leftrightarrow j$ is potentially true. This will help prune the search-space of possible correspondences. Out of this reduced search-space, we shall later extract the correspondences which give the minimum uncertainty volume of registration.

A. Some Tests for Pruning the Correspondence Search Space

1) Size Similarity Test: As shown in Appendix A, the inverse of the 4×4 covariance matrix of the plane-parameters is proportional to the number of points in the plane. Therefore, the determinant of this matrix is proportional to the fourth-power of the size of the point-cloud and it is also a function of the associated certainty of the points. One way to restrict the correspondence search-space is to discard pairs for which the log of the ratio of this value exceeds some threshold, i.e., we discard the possibility that $i \leftrightarrow j$ if

$$\left|\log|{}^{\ell}\mathbf{C}_i^{+}|_{+} - \log|{}^{r}\mathbf{C}_i^{+}|_{+}\right| > \bar{L}_{\text{det}} \tag{38}$$

2) Given Translation Agreement Test: If an estimate of the translation ${}^{\ell}_{r}\mathbf{t}_e$ is given along with its covariance $\mathbf{C_{tt}}$ and a potential correspondence ${}^{\ell}\mathcal{P}_i \leftrightarrow {}^{r}\mathcal{P}_j$, (4b) can be used again to form an error metric

$$\chi^2_{t,e} \triangleq \frac{({}^{\ell}\hat{\mathbf{n}}_i \cdot {}^{\ell}_{r}\mathbf{t}_e - {}^{\ell}d_i + {}^{r}d_j)^2}{{}^{\ell}\mathbf{D}_{i,dd} + {}^{r}\mathbf{D}_{j,dd} + {}^{\ell}_{r}\mathbf{t}_e^{\top}\,{}^{\ell}\mathbf{D}_{i,\hat{\mathbf{n}}\hat{\mathbf{n}}}\,{}^{\ell}_{r}\mathbf{t}_e + {}^{\ell}\hat{\mathbf{n}}_i^{\top}\mathbf{C_{tt}}\,{}^{\ell}\hat{\mathbf{n}}_i} \quad (39)$$

If $\chi^2_{t,e} > \bar{\chi}^2_t$, the hypothesis that the potential correspondence agrees with the given translation is rejected.

3) Odometry Rotation Agreement Test: Similarly, if the roll (ψ_{o}), pitch (θ_{o}), yaw (ϕ_{o}) angles for ${}^{\ell}_{r}\mathbf{R}$ according to odometry are given along with their covariance matrix $\mathbf{C}_{\phi,\theta,\psi}$, we can use them to eliminate potential correspondence pairings which cause a gross disagreement with the odometry values. Suppose we have selected two pairs of corresponding normals ${}^{\ell}\hat{\mathbf{n}}_i$, ${}^{r}\hat{\mathbf{n}}_i$, $i = 1, 2$. If these pairs are non-parallel, the rotation is fully determined and can be computed as ${}^{\ell}_{r}\mathbf{R}$ from the results of Sec. II-C. From ${}^{\ell}_{r}\mathbf{R}$, the corresponding roll (ψ), pitch (θ), yaw (ϕ) can be extracted. Then the Mahalanobis distance is computed as

$$\mathbf{e}_r \triangleq \begin{bmatrix} \phi_{\mathrm{o}} - \phi \\ \theta_{\mathrm{o}} - \theta \\ \psi_{\mathrm{o}} - \psi \end{bmatrix}, \qquad \chi^2_{R,o} \triangleq \mathbf{e}_r^{\top}\mathbf{C}^{-1}_{\phi,\theta,\psi}\mathbf{e}_r. \quad (40)$$

If $\chi^2_{R,o} > \bar{\chi}^2_{R,o}$, we can reject the hypothesis that the selected correspondences agree with the odometry.

In this article, we use odometry only as a switch to restrict the search space for correspondences. Within this space, however, there is no *attraction* towards the odometry estimation. Thus, even if actual odometry is unavailable but some reasonable bounds for pose change are known, they can still be used.

4) Plane-patch Overlap Test: In certain rare cases, there might be more than one patch with very close $(\hat{\mathbf{n}}, d)$ values – imagine two picture frames on the same wall. Disambiguating among them is usually not necessary for relative pose estimation, if there are planes perpendicular to these patches. If, however, no such plane is found, then there is uncertainty about the translation parallel to the infinite plane containing the two patches. In such a case, the only way to disambiguate the patch-correspondences across two views is to consider whether one planar patch overlaps with another after application of rotation and translation. If we calculate such estimates for two pairs of supposedly corresponding planes i, j, then we can compute an overlap translation estimate for each of the pairs using the method of Sec. II-E. Two such estimates will be considered consistent if

$$\mathbf{e}_{ij} \triangleq {}^{\ell}_{r}\mathbf{t}_i - {}^{\ell}_{r}\mathbf{t}_j, \quad \bar{\chi}^2_{\mathrm{ovlp}} \geq \mathbf{e}_{ij}^{\top}({}^{\ell}_{r}\mathbf{C}_{i,\mathbf{t}} + {}^{\ell}_{r}\mathbf{C}_{j,\mathbf{t}})^{-1}\mathbf{e}_{ij}. \quad (41)$$

5) Cross-Angle Test: In a rigid motion, the angles between normals in one sample should be the same as the corresponding angles in the next. Suppose we are given planes normals and covariances $\{{}^{\ell}\hat{\mathbf{n}}_{i_1}, {}^{\ell}\mathbf{D}_{i_1,\hat{\mathbf{n}}\hat{\mathbf{n}}}\}, \{{}^{\ell}\hat{\mathbf{n}}_{j_1}, {}^{\ell}\mathbf{D}_{j_1,\hat{\mathbf{n}}\hat{\mathbf{n}}}\}$ in the left scene and $\{{}^{r}\hat{\mathbf{n}}_{i_2}, {}^{r}\mathbf{D}_{i_2,\hat{\mathbf{n}}\hat{\mathbf{n}}}\}, \{{}^{r}\hat{\mathbf{n}}_{j_2}, {}^{r}\mathbf{D}_{j_2,\hat{\mathbf{n}}\hat{\mathbf{n}}}\}$ in the right, such that potentially ${}^{\ell}\hat{\mathbf{n}}_{i_1} \leftrightarrow {}^{r}\hat{\mathbf{n}}_{i_2}$ and ${}^{\ell}\hat{\mathbf{n}}_{j_1} \leftrightarrow {}^{r}\hat{\mathbf{n}}_{j_2}$. Ideally,

$${}^{\ell}v_{i_1,j_1} \triangleq {}^{\ell}\hat{\mathbf{n}}_{j_1}^{\top}\,{}^{\ell}\hat{\mathbf{n}}_{i_1}, \quad {}^{r}v_{i_2,j_2} \triangleq {}^{r}\hat{\mathbf{n}}_{j_2}^{\top}\,{}^{r}\hat{\mathbf{n}}_{i_2}, \quad {}^{\ell}v_{i_1,j_1} \approx {}^{r}v_{i_2,j_2} \quad (42)$$

The variance of ${}^{\ell}v_{i_1,j_1}$ is ${}^{\ell}\sigma^2_{i_1,j_1} \approx {}^{\ell}\hat{\mathbf{n}}_{j_1}^{\top}\,{}^{\ell}\mathbf{D}_{i_1,\hat{\mathbf{n}}\hat{\mathbf{n}}}\,{}^{\ell}\hat{\mathbf{n}}_{j_1} + {}^{\ell}\hat{\mathbf{n}}_{i_1}^{\top}\,{}^{\ell}\mathbf{D}_{j_1,\hat{\mathbf{n}}\hat{\mathbf{n}}}\,{}^{\ell}\hat{\mathbf{n}}_{i_1}$ and ${}^{r}\sigma^2_{i_2,j_2}$ is similarly defined. A distance metric can be defined as

$$\chi^2_{\times} \triangleq \frac{({}^{\ell}v_{i_1,j_1} - {}^{r}v_{i_2,j_2})^2}{{}^{\ell}\sigma^2_{i_1,j_1} + {}^{r}\sigma^2_{i_2,j_2}}. \quad (43)$$

If $\chi^2_{\times} > \bar{\chi}^2_{\times}$, we can reject the assumed correspondences. This is our main test for geometric consistency.

6) Parallel Consistency Test: Given two potentially corresponding pairs ${}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}$ and ${}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}$, such that either both ${}^{\ell}\hat{\mathbf{n}}_{i_1} \cdot {}^{\ell}\hat{\mathbf{n}}_{j_1} \approx 1$ and ${}^{r}\hat{\mathbf{n}}_{i_2} \cdot {}^{r}\hat{\mathbf{n}}_{j_2} \approx 1$, or both ${}^{\ell}\hat{\mathbf{n}}_{i_1} \cdot {}^{\ell}\hat{\mathbf{n}}_{j_1} \approx -1$ and ${}^{r}\hat{\mathbf{n}}_{i_2} \cdot {}^{r}\hat{\mathbf{n}}_{j_2} \approx -1$, i. e. the planes are parallel or anti-parallel. Then on using (4b), we should also have $\delta \triangleq ({}^{\ell}d_{i_1} - {}^{r}d_{i_2}) \mp ({}^{\ell}d_{j_1} - {}^{r}d_{j_2}) \approx 0$, where the $+$ is taken for the antiparallel case. This can be checked by computing

$$\chi^2_{\delta} \triangleq \frac{\delta^2}{{}^{\ell}\mathbf{D}_{i_1,dd} + {}^{\ell}\mathbf{D}_{j_1,dd} + {}^{r}\mathbf{D}_{i_2,dd} + {}^{r}\mathbf{D}_{j_2,dd}}. \quad (44)$$

The consistency hypothesis can be rejected if the $\chi^2_{\delta} > \bar{\chi}^2_{\delta}$.

B. The Minimally Uncertain Maximal Consensus (MUMC) Algorithm

Given an average of N planes per view, there are $(\mathsf{N}+1)!$ possible correspondences, if we include the case when a plane in one view is not present in the other. We can naïvely try all of these correspondences, possibly using the extra tests of Sec. III-A to cut-down the search space, and choose the one with the maximum overall consistency. Typically, the number of planes is much less than the number of points used to compute them. Therefore such an exhaustive search may work for a small value of N. We present next a novel algorithm which aims to find the most consistent set with respect to rotation and translation. It uses the fact that to uniquely determine rotation one needs to know at least two non-parallel plane correspondence pairs, and for translation at least three. Non-parallel planes' correspondences have rotation information and parallel planes' correspondences have only translation information.

The algorithm initially finds all consistent 2 pairs of correspondences (i.e. quadruplets) in its preprocessing step using the tests devised in the last sections. In the main search step, each of these pairs are considered in turn and their largest rotation and translation consensus sets are built. For each of these consensus sets, the least-squares rotation and translation are determined, along with the volume of uncertainty given by the pseudo-determinant of the covariance matrix of the pose-registration. The consensus set of a certain minimum size N_C (explained later) which has the minimum value of the uncertainty volume is selected as the chosen set of correspondences.

Although a formulation of RANSAC for plane-matching itself would have been novel enough since its robustly proven application in this field is currently lacking in the literature, we have added further improvements to the procedure. Unlike RANSAC, there is no random search, and instead of a greedy maximum consensus, a minimum uncertainty volume approach is taken. We have found this approach to be much more stable in practice than one based solely on the size of the consensus set. The uncertainty volume automatically decreases if more *consistent* correspondence pairs are added to the set. If, however, a pair is added which merely passes the consistency thresholds but is not overall consistent, the uncertainty volume increases. This also removes the necessity of over-tuning the thresholds in the consistency tests.

Before we go on to the algorithm, the basic notation used in this section is summarized below:

$\#\Omega$	Size of the set or list Ω.
$\omega.a$	Member a of a tuple ω.
${}^{\ell}\mathcal{P}_i \upharpoonleft\upharpoonright {}^{\ell}\mathcal{P}_j$	Planes i and j in the ℓ set are parallel.
${}^{r}\mathcal{P}_i \upharpoonleft\downharpoonright {}^{r}\mathcal{P}_j$	Planes i and j in the r set are antiparallel.
${}^{\ell}\mathcal{P}_i \nparallel {}^{\ell}\mathcal{P}_j$	Planes are neither parallel nor antiparallel.
${}^{\ell}\mathcal{P}_i \leftrightarrow {}^{r}\mathcal{P}_j$	${}^{\ell}\mathcal{P}_i$ and ${}^{r}\mathcal{P}_j$ correspond to each other.

A test for parallelism can be devised by testing for nearness of the dot product to unity within some threshold or a statistical measure can be derived along the lines of (43).

We consider the following consistency requirements, in increasing order of complexity:

1) A potential correspondence ${}^{\ell}\mathcal{P}_i \leftrightarrow {}^{r}\mathcal{P}_j$ is called *1-consistent* if it passes the size similarity test (38) and the translation agreement test (III-A2) for odometry, if available.
2) Two potential correspondences ${}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}$ and ${}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}$ are called
   a) *Rotation 2-consistent*: if ${}^{\ell}\mathcal{P}_{i_1} \nparallel {}^{\ell}\mathcal{P}_{j_1}$ and ${}^{r}\mathcal{P}_{i_2} \nparallel {}^{r}\mathcal{P}_{j_2}$, then a unique 3D rotation estimate ${}^{\ell}_{r}\mathbf{R}$ can be computed from the assumption that ${}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}$ and ${}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}$. If they pass the cross-angle test (III-A5), the overlap test of Sec. III-A4, and also the test of (40), if odometry is available, then the pair is rotation 2-consistent.
   b) *Translation 2-consistent*: if either ${}^{\ell}\mathcal{P}_{i_1} \upharpoonleft\upharpoonright {}^{\ell}\mathcal{P}_{j_1}$ and ${}^{r}\mathcal{P}_{i_2} \upharpoonleft\upharpoonright {}^{r}\mathcal{P}_{j_2}$, or ${}^{\ell}\mathcal{P}_{i_1} \upharpoonleft\downharpoonright {}^{\ell}\mathcal{P}_{j_1}$ and ${}^{r}\mathcal{P}_{i_2} \upharpoonleft\downharpoonright {}^{r}\mathcal{P}_{j_2}$, and in addition the two pairs pass the parallel consistency test of Sec. III-A6.

The first step of the algorithm, shown in Algo. 1, consists of collecting all non-parallel rotationally 2-consistent correspondence pairs in set $\Omega_{\nparallel}$, and all parallel and translationally 2-consistent pairs in set $\Omega_{\parallel}$. Both sets consist of elements which are tuples of format $\langle\{{}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}, {}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}\}, \chi^2\rangle$. For notational convenience and due to symmetry, the *order* of the two pairs in the tuple is not important. Hence, the aforementioned tuple is the same as the tuple $\langle\{{}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}, {}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\}, \chi^2\rangle$. We can define the following:

1) A set of interesting pairs $\mathcal{K}$, such that the correspondence ${}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2} \in \mathcal{K}$ if it exists in any of the tuples in $\Omega_{\parallel}$ or $\Omega_{\nparallel}$.
2) A set of correspondences which are translation 2-consistent to a given correspondence.

$$\left[{}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\right]_{\parallel} \triangleq \Big\{\langle {}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}, \chi^2\rangle \mid \langle\{{}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}, {}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\}, \chi^2\rangle \in \Omega_{\parallel}\Big\} \quad (45)$$

The reader is reminded of the order irrelevance within a tuple due to symmetry.

3) A set of correspondences which are not parallel or anti-parallel to a given correspondence, but are *rotation 2-consistent* with it.

$$\left[{}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\right]_{\nparallel} \triangleq \Big\{\langle {}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}, \chi^2\rangle \mid \langle\{{}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}, {}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\}, \chi^2\rangle \in \Omega_{\nparallel}\Big\} \quad (46)$$

Algorithm 1: MUMC Preprocessing Step: Find all consistent correspondence pairs

input : Left planes indexed list ${}^{\ell}\mathcal{P} = \{\langle {}^{\ell}\hat{\mathbf{n}}_i, {}^{\ell}d_i, {}^{\ell}\mathbf{C}_i\rangle \mid i = 1 \ldots \mathsf{N}_\ell\}$, and right planes indexed list ${}^{r}\mathcal{P} = \{\langle {}^{r}\hat{\mathbf{n}}_i, {}^{r}d_i, {}^{r}\mathbf{C}_i\rangle \mid i = 1 \ldots \mathsf{N}_r\}$.
output: Sets $\Omega_{\parallel}$, $\Omega_{\nparallel}$.

```
Sort planes in descending order of evidence, i.e.,
  |^s C_i^+|_+, s = r/ℓ and take a certain top F_t%.
Compute decoupled covariances using the procedure in
  Sec. II-B. Make a list L of all unique and 1-consistent
  potentially corresponding pairs
  (^ℓP_i ↔ ^rP_j), i = 1...N_ℓ, j = 1...N_r. Obviously,
  #L ≤ N_ℓ × N_r.
for k = 1...#L − 1 do
    (^ℓP_{k1} ↔ ^rP_{k2}) ← L[k]
    for i = k + 1...#L do
        (^ℓP_{i1} ↔ ^rP_{i2}) ← L[i]
        if L[k], L[i] translation 2-consistent then
            Compute χ²_δ from parallel consistency test
              (44)
            If the above test passed, add
              ⟨^ℓP_{k1} ↔ ^rP_{k2}, ^ℓP_{i1} ↔ ^rP_{i2}, χ²_δ⟩ to Ω_∥
        else if L[k], L[i] rotation 2-consistent then
            Compute χ²_× from cross-angle consistency
              test (43)
            If the above test passed, add
              ⟨^ℓP_{k1} ↔ ^rP_{k2}, ^ℓP_{i1} ↔ ^rP_{i2}, χ²_×⟩ to Ω_∦
        end
    end
end
```

The second step, shown in Algo. 2 considers each member of $\mathcal{K}$ and finds the member with the minimum uncertainty in the determination of ${}^{\ell}_{r}\mathbf{t}$ and ${}^{\ell}_{r}\mathbf{R}$. We first give an overview skeleton of the algorithm in Algo. 2. Lines 1, 5, 6, 7, and 9 are explained afterwards.

Explanation of Line 1: N_C is set to 4, because the minimum number of corresponding pairs required to uniquely

Algorithm 2: MUMC Main Search: Finding minimally uncertain maximal consensus

input : $\Omega_{\parallel}$ and $\Omega_{\nparallel}$ from the output of Algo. 1.
output: The resolved correspondences and the computed transform.

1 Set Minimum required consensus size $\mathsf{N}_C \leftarrow 4$.
2 Initialize the potential solution list $\mathcal{W} \leftarrow \varnothing$.
3 Initialize selected solution tuple $\bar{\omega} \leftarrow \varnothing$.
4 **for** $\forall\, {}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2} \in \mathcal{K}$ **do**
5   Find the maximal rotation consensus set $\Gamma_{\nparallel}$ in $\left[{}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\right]_{\nparallel}$.
6   Find the maximal parallel consensus set $\Gamma_{\parallel}$ in $\left[{}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\right]_{\parallel}$.
7   Find the maximal set Γ_t in $\Gamma_{\nparallel}$ which is translationally consistent.
8   Append $\Gamma \leftarrow \Gamma_t \cup \Gamma_{\parallel}$.
9   Using Γ, and results of Sec. II, compute the rotation ${}^{\ell}_{r}\check{\mathbf{q}}$ and its covariance ${}^{\ell}_{r}\mathbf{C}_{\check{\mathbf{q}}\check{\mathbf{q}}}$, and translation ${}^{\ell}_{r}\mathbf{t}$ and its covariance ${}^{\ell}_{r}\mathbf{C}_{\mathbf{tt}}$.
10   Compute uncertainty-volume metric $\alpha = |{}^{\ell}_{r}\mathbf{C}_{\mathbf{tt}}| \times |{}^{\ell}_{r}\mathbf{C}_{\check{\mathbf{q}}\check{\mathbf{q}}}|_{+}$
11   Assign tuple $\omega_i \leftarrow \left\langle \alpha, \Gamma, {}^{\ell}_{r}\check{\mathbf{q}}, {}^{\ell}_{r}\mathbf{C}_{\check{\mathbf{q}}\check{\mathbf{q}}}, {}^{\ell}_{r}\mathbf{t}, {}^{\ell}_{r}\mathbf{C}_{\mathbf{tt}} \right\rangle$ and append to $\mathcal{W}$.
12   **if** $\#(\omega_i.\Gamma) \geq \mathsf{N}_C$ **or** $\#(\bar{\omega}.\Gamma) \leq \mathsf{N}_C$ **then**
13   **if** $\omega_i.\alpha < \bar{\omega}.\alpha$ **then**
14             $\bar{\omega} \leftarrow \omega_i$
15   **end**
16   **end**
17 **end**
18 The tuple $\bar{\omega}$ is the chosen solution. Other potential solutions are in the list $\mathcal{W}$.

compute translation is 3. Therefore, we can speak of "consensus" only if we have ≥ 4 pairs in agreement.

Explanation of Line 5: We first note that a rotation ${}^{\ell}_{r}\mathbf{R}$ can be uniquely found by freezing the correspondences ${}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}$ and another one $\langle {}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}, \chi^2_{\times} \rangle \in \left[{}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\right]_{\nparallel}$. A third correspondence $\langle {}^{\ell}\mathcal{P}_{k_1} \leftrightarrow {}^{r}\mathcal{P}_{k_2}, \chi^2_{\times} \rangle \in \left[{}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\right]_{\nparallel}$ is called rotationally consistent with the two frozen correspondences, if ${}^{\ell}\hat{\mathbf{n}}_{k_1} \cdot ({}^{\ell}_{r}\mathbf{R}\, {}^{r}\hat{\mathbf{n}}_{k_2}) \approx 1$. A test analogous to (43) or even a threshold based test can be used here. We have used a lower threshold of 0.9980 for all sensors. The set of all rotationally consistent correspondences to the frozen pair is denoted $\Omega_R\left({}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}, {}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}\right)$ and its overall $\sum \chi^2_{\times}$ can be evaluated by summing up the $\chi^2_{\times}$ values of all tuples belonging to the set. We then find the largest such set denoted $\Omega^{\star}_R\left({}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\right)$ by considering all ${}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}$, where, if two sets are of equal size, then the one having lower overall $\sum \chi^2_{\times}$ is taken. The set $\Omega^{\star}_R\left({}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\right)$ may still contain non-unique correspondences, i.e. some plane in the ℓ set may be mapped to more than one other planes in the r set and vice versa. The uniqueness problem is solved by sorting all $\langle {}^{\ell}\mathcal{P}_{k_1} \leftrightarrow {}^{r}\mathcal{P}_{k_2}, \chi^2_{\times} \rangle \in \Omega^{\star}_R\left({}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\right)$ in increasing order of $\chi^2_{\times}$ and then fixing correspondences by traversing the sorted list. If ${}^{\ell}\mathcal{P}_{k_1}$ can be matched with more than one plane in the r set, then the pairing with the lower $\chi^2_{\times}$ is automatically chosen. Similar reasoning applies if ${}^{r}\mathcal{P}_{k_2}$ can be matched with more than one plane in the ℓ set. The reduced and consistent unique set of correspondences is the maximal rotation consensus set $\Gamma_{\nparallel}$.

TABLE I
SENSOR CHARACTERISTICS

	Swiss-ranger	USF Odetics	ALRF
FOV $h \times v$	$47^\circ \times 39^\circ$	$60^\circ \times 60^\circ$	$270^\circ \times 180^\circ$
Resolution $h \times v$	176×144	128×128	541×361
Total points	$25,344$	$16,384$	$195,301$
Range (mm)	$7,500$	unknown	$> 16,000$

Explanation of Line 6: The set $\left[{}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\right]_{\parallel}$ may also contain non-unique correspondences, i.e. some plane in the ℓ set may be mapped to more than one other planes in the r set and vice versa. The uniqueness problem is solved by sorting all $\langle {}^{\ell}\mathcal{P}_{k_1} \leftrightarrow {}^{r}\mathcal{P}_{k_2}, \chi^2_{\delta} \rangle \in \left[{}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}\right]_{\parallel}$ in increasing order of χ^2_{δ} and then fixing correspondences by traversing the sorted list. If ${}^{\ell}\mathcal{P}_{k_1}$ can be matched with more than one plane in the r set, then the pairing with the lower χ^2_{δ} is automatically chosen. Similar reasoning applies if ${}^{r}\mathcal{P}_{k_2}$ can be matched with more than one plane in the ℓ set. The reduced and consistent unique set of correspondences is the maximal parallel consensus set $\Gamma_{\parallel}$.

Explanation of Line 7: Here we find the largest subset of $\Gamma_{\nparallel}$ which is translationally consistent. Unlike rotation, here we need to freeze three correspondences to compute translation. The first ${}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}$ is given; we take two others ${}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}, {}^{\ell}\mathcal{P}_{k_1} \leftrightarrow {}^{r}\mathcal{P}_{k_2} \in \Gamma_{\nparallel}$. Using these we can compute ${}^{\ell}_{r}\mathbf{t}$. We then find all correspondences ${}^{\ell}\mathcal{P}_{p_1} \leftrightarrow {}^{r}\mathcal{P}_{p_2} \in \Gamma_{\nparallel}$ which pass the given translation agreement test of Sec. III-A2 for ${}^{\ell}_{r}\mathbf{t}$. For each such test evaluation, we also get a $\chi^2_{t,e}$ value. We denote the set of all such translationally consistent correspondences as $\Omega_t\left({}^{\ell}\mathcal{P}_{i_1} \leftrightarrow {}^{r}\mathcal{P}_{i_2}, {}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}, {}^{\ell}\mathcal{P}_{k_1} \leftrightarrow {}^{r}\mathcal{P}_{k_2}\right)$. This set is also associated with an overall $\sum \chi^2_{t,e}$ value, which is the sum of $\chi^2_{t,e}$ for all the correspondences in it. The maximal such set for all ${}^{\ell}\mathcal{P}_{j_1} \leftrightarrow {}^{r}\mathcal{P}_{j_2}$ and ${}^{\ell}\mathcal{P}_{k_1} \leftrightarrow {}^{r}\mathcal{P}_{k_2}$ is termed Γ_t— where, if two sets are of equal size, then the one with the overall lesser value of $\sum \chi^2_{t,e}$ is preferred.

Explanation of Line 9: This line shows that we reuse the results of least-squares registration estimation for known-correspondences from Sec. II, including the associated rotation and translation covariances, to evaluate the goodness of the assumed correspondence-set Γ.

IV. RESULTS

We present results for three sensors 1) Swiss-Ranger (SR) 2) USF Odetics, and 3) A SICK S300 laser range finder, actuated (pitched) using a servo (ALRF). The characteristics of these sensors are listed in Table I, which shows that the first two are small field of view (FOV) sensors and ALRF has a large FOV. The computation times mentioned in this paper are for an AMD Turion 2×64, 1.6 GHz laptop with 960 MB RAM running OpenSUSE 10.3 O/S.

TABLE II
MUMC PARAMETERS (UNITS MM AND RADIANS). S.D. STANDS FOR SCENE-DEPENDENT

Parameter	Swiss-ranger	USF Odetics	ALRF
$F_t\%$	≥ 50	≥ 80	15 – S.D.
$\epsilon_1, \bar{c}$	$10^{-7}, 25-50$	$10^{-7}, 5-10$	$10^{-7}, 50-70$
$\bar{L}_{\text{det}}$	$4-20$	$10-20$	$10-20$
$\bar{\chi}^2_{\text{ovlp}}$	$2.5-4$	$2.5-4$	$2-4$
$\bar{\chi}^2_{\times}$	$(1-100)\times 10^5$	$(1-100)\times 10^5$	$(3-10)\times 10^5$
$\bar{\chi}^2_{\delta}$	$10-300$	$200-300$	$10-200$
$\bar{\chi}^2_{t,e}$	$\chi^2_{1,1.5\%} = 3.84$	$\chi^2_{1,1.5\%}$	$\chi^2_{1,1.5\%}$
κ	6	6	6

TABLE III
COMPUTATION TIMES (SECONDS) FOR PRE-PROCESSING STEP FOR SWISS-RANGER.

Scan Nr.	Region Growing	Nr. Patches	Polygonization
1	0.52	39	0.38
2	0.38	25	0.35
3	0.38	25	0.34
4	0.4	13	0.41
5	0.39	15	0.42
6	0.51	23	0.52

A. Selection of Parameters

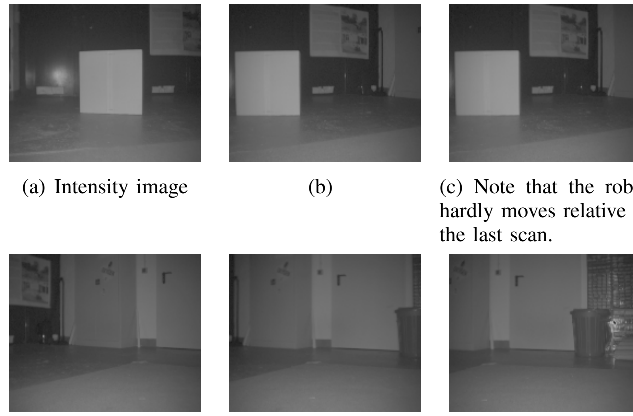
Refer to Table II for parameter and threshold values used by us for various sensors. For most parameters, we have provided an approximate range of values which work. In most cases, the ranges for different sensors for a given parameter overlap. When a simple chi-square test from standard tables suffices, only a single value is given. For example, in the last row, $\chi^2_{1,1.5\%}$ means the χ^2 value for 1 d.o.f. at a significance level of 1.5% [32].

Obviously, these parameters depend on the the error model considered for the sensor while extracting the planes. For each sensor, a parameter-tuning is required once. The observed values of some manually registered planes can be used to arrive at these thresholds. Lower values of thresholds are better because they provide more initial filtering and thus speed-up the algorithm.

The size-similarity threshold $\bar{L}_{\text{det}}$ and overlap threshold $\bar{\chi}^2_{\text{ovlp}}$ also depend on the amount of occlusion present in the scene and the rotational and translational distance at which the samples are taken. They need to be reduced when the occlusion is severe or the odometry is not available, otherwise false matches may result. The filtering F_t is selected so that for low FOV sensors most of the planes are retained, while for high FOV sensors, many low evidence planes can be filtered out to pick up speed.

B. Swiss-Ranger Rotation Data-set

A test sequence of overlapping views, taken from an on-board Swiss-ranger (SR), is shown in Fig. 2. This data-set is particularly interesting because of the relatively large rotation of the robot between every pair of successive views. The computation times for planar-patches extraction using region-growing and for polygonization is shown in Table III.



(a) Intensity image (b) (c) Note that the robot hardly moves relative to the last scan.

(d) (e) (f)

Fig. 2. The intensity images obtained from the Swiss-Ranger. The planes were extracted from the corresponding range images. The MUMC result is shown in Fig. 3.

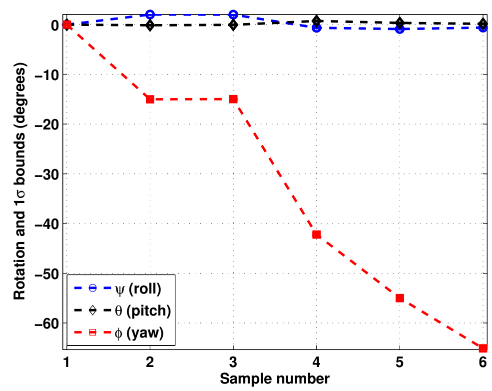
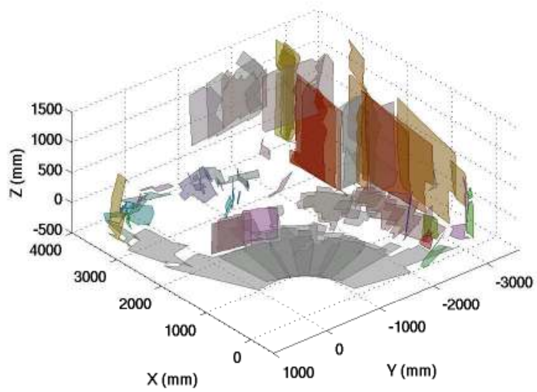


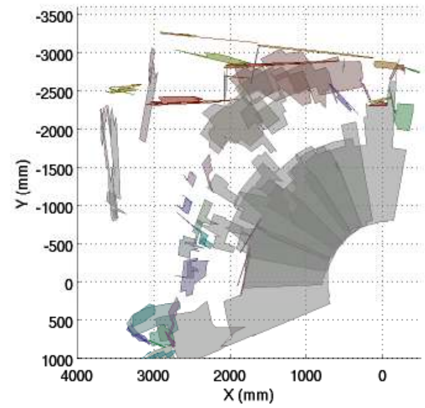
Fig. 5. Computed robot pose and uncertainties for the data-set of Fig. 3. Only rotation is shown because the translation is insignificantly small for this dataset. The rotational uncertainties are too small to be visible in the figure.

The results of scan-matching using the MUMC algorithm are shown in Figs. 3(c) and 3(d). We emphasize that no odometry was used for the results shown in this figure to test for robustness. Fig. 4 shows the result of plane-fusion as described in [17] for the matched planes, followed by projection of the points on the fused plane. The evolution of the robot pose and its uncertainties is shown in Fig. 5. The uncertainties are small but cumulatively increase. This can be avoided by embedding the approach in a SLAM framework, where the overall uncertainty can decrease when certain features (planes) are viewed again in several successive samples.

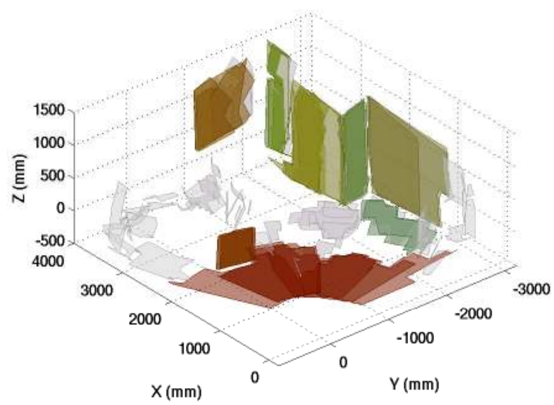
1) Implementation of point-to-point (P-P) ICP: The P-P ICP algorithm [8] for aligning point clouds has found wide usage in the 3D mapping community [9], [6] for estimating robot movement between two successive time-instants. For comparison, we have coded our own implementation in C++. The code uses an optimized kd-tree [39] for fast nearest-neighbor search. The mean-square error e_k at iteration k reduces monotonically with iteration k. The convergence criterion used was $(e_k - e_{k+1})/e_k < 0.0001$ which is independent of the size of the point-cloud.



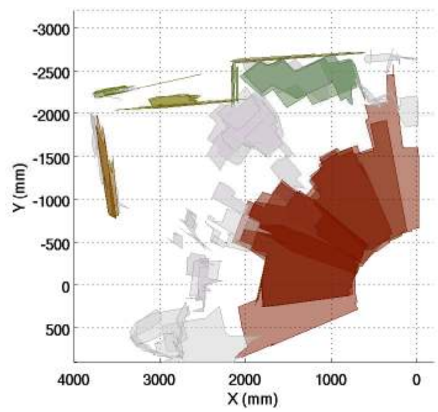
(a) Odometry based pose estimation with all planes drawn transparently: front view. The plane color is a function of its normal direction and is not related to any matching.



(b) Odometry based pose estimation with all planes drawn transparently: top view

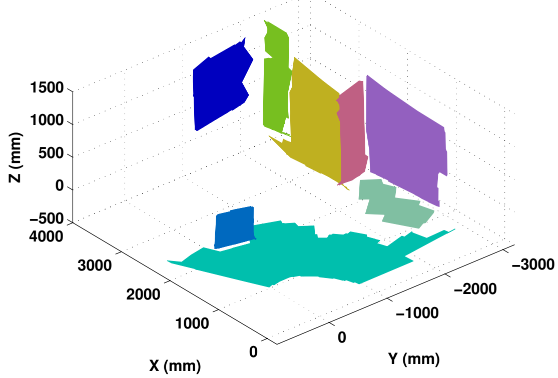


(c) Planes matched and aligned using MUMC: front view. Corresponding matched planes are drawn with the same color. Unmatched and filtered out planes are shown grayed out.

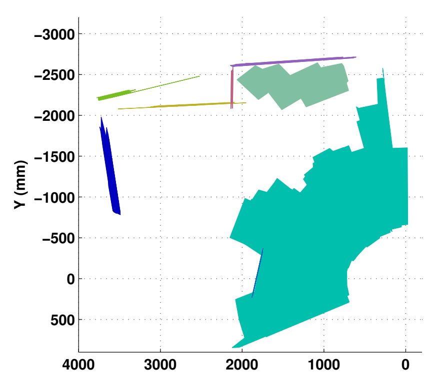


(d) Planes matched and aligned using MUMC: top view. Unmatched and filtered out planes are shown grayed out.

Fig. 3. Results of MUMC for matching planes from the overlapping views from a Swiss-Ranger corresponding to Fig. 2(a) to 2(f). Planes are drawn transparently to show overlap. The map based on pure odometry is shown in Figs. 3(a) and 3(b). Result of MUMC algorithm is shown in Figs. 3(c) and 3(d). For the result shown in Figs. 3(c) and 3(d), as a check, *no odometry was supplied to the MUMC algorithm*. The algorithm aligned most planes properly.



(a) Fused planes: front view.



(b) Fused planes: top view.

Fig. 4. The result of applying plane fusion based on the methodology described in [17] on the matching results shown in Fig. 3(c) and 3(d). Rotations were computed using the faster method of Sec. II-C which uses isotropic uncertainties.

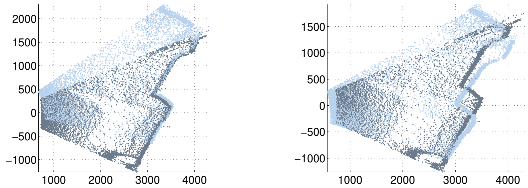
2) Implementation of point-to-plane (P-L) ICP: Our implementation of (P-L) ICP [10], [11] uses a kd-tree [39] to find intersection of a line and a surface. A sliding window of size 15×15 is used to estimate *local* normals needed for the algorithm. The cost-function [10, Eq. (11)] has no closed-form solution. Therefore, it is minimized using the method of linearization recommended in [10]. The convergence threshold was taken as $\epsilon_e = 10$. Unlike P-P ICP, the convergence of P-L ICP is not monotonic because the number of found correspondences varies across iterations.

3) Implementation of 3D NDT: 3D NDT [5], [40] is a voxel based approach akin to occupancy grids, except that a probability distribution rather than a single occupancy value is defined for each cell. The authors of [5] have kindly provided us access to their code, and they have also tuned the parameters of NDT for our datasets. These parameters for the Swiss-Ranger data-set were: bin spacing 2.0, with iterative-split 0.1.

4) Comparison of Algorithms: Point-to-point ICP is robust with respect to large perturbations *if the point set itself has not changed*. However, as Fig. 6 shows, P-P ICP becomes brittle if applied to two successive views with moderate pose change. In the example, it was a rotation of about $12°$. This lack of robustness occurs due to the points in one view which are absent in the other. Fig. 6(a) shows the solution given by the MUMC algorithm, which shows good alignment. Even if we initialize the ICP with this solution, it still converges to the result in Fig. 6(b). Thus, the real solution is not even a local minimum with respect to the P-P ICP cost function. In fact, it converges to a wrong solution for all except one of the pairs of view shown in Fig. 2. The exception pair is from Figs. 2(b) and 2(c), between which the robot does not move, although the odometry says it does, and the only difference in the point-clouds is due to the sensor noise.

This lack of robustness is, of course, a function of the field of view (FOV) and range of the 3D sensor employed. Higher values of these parameters will effectively increase the pose-difference convergence radius of ICP. Nüchter et al [6] used a servo-operated laser range finder (LRF) in a stop-and-go fashion. Their sensor had the FOV of $180°(\mathrm{h}) \times 120°(\mathrm{v})$. The range was not mentioned, although the SICK LRF employed is known to have a range of about 30 m. Such a high FOV and range allowed them to take measurements every 2.5 m. On the other hand, the Swiss-Ranger has a much smaller FOV and range. As shown in Sec. IV-B, having a sensor with a higher FOV only helps the MUMC algorithm, since in this case the number of overlapping planes within successive views increases.

The comparison of computation times is summarized in Table IV. A scan-matching success is marked with $\surd$, failure with $\times$, and if the matching was only *partially* successful (e.g. in Fig. 6(b), where the rotation is more or less correct), it is marked with a 'P'. The heading '100 %' in P-P and P-L ICP shows that all the points were taken. The heading '100 %' in MUMC is the parameter $F_t\%$ value and shows that all planes were considered in matching. On the other hand, '50 %' means that the planes were sorted based on their evidence and only the top half were considered for matching. The matching results remain unaffected, though the computation time is drastically reduced. For 3D NDT 100% of the points are supplied, though due to the grid discretization, the point-cloud is effectively compressed.



(a) Point clouds initialized with the MUMC solution. Note the good alignment.

(b) Final convergent solution from initialization using MUMC solution of Fig. 6(a).

Fig. 6. Registration done using point-to-point ICP. Top view of point-cloud pair corresponding to successive samples of Figs. 2(d) and 2(e). Point-cloud sub-sampled by 3 for better visualization.

As Table IV makes clear, both P-P and P-L ICP can be made faster and more accurate by providing odometry estimates. P-P ICP was overall least successful for this data-set. Both P-L ICP and 3D NDT were unable to match pair $3, 4$ without odometry. In this pair, a predominant plane suddenly goes out of view. By contrast, MUMC was able to process all scan-pairs without being provided any odometry or prior estimate. Even if we add the time for plane-extraction using region-growing (Table III) to the time taken by MUMC, it still remains the fastest method among the four. We note that the polygonization step is only required for visualization, and that, except for the first pair, the matching of two scans requires the planes extraction by region-growing of only one new scan. The relatively higher time of processing for the first pair by MUMC for the '100 %' case is because of the large number (39) of patches in the first scan, most of which were of low evidence. The columns with heading # show the number of corresponding pairs found.

We can thus conclude that the MUMC approach is both faster and more robust than both variants of ICP. As compared to 3D NDT, MUMC is more robust with a similar computation time when no evidence based sorting is used. When the planes are sorted based on their evidence and only the top half are considered for matching, the matching results remain unaffected and the computation time for MUMC is drastically reduced, making it by far the fastest.

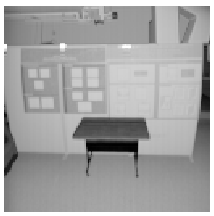
Another major advantage is in terms of storage requirements. A plane summarizes the information from a lot of spatial points. Storing these planes in a graph node in a GraphSLAM-like approach is thus much more storage-efficient than storing whole point-clouds. This continues to be valid, even if we store the polygonized boundaries of the points belonging to a plane, along with the parameters of the infinite plane. The only disadvantage of MUMC is that it requires planar features, i.e., indoors and urban outdoor scenarios, though we also had surprisingly good results in quite unstructured outdoor environments [16].

C. USF Odetics LADAR Ego-Motion Data-set r1

The USF range-image is a popular data-set, which has been online now for over a decade. We tested our algorithm with the ego-motion data-subset r1 recorded using the Odetics LADAR

TABLE IV
SR DATA-SET: COMPARISON OF COMPUTATION TIME (SEC.) AND MATCHING-SUCCESS FOR DIFFERENT ALGORITHMS. MUMC WAS THE ONLY MATCHER TO SUCCEED FOR ALL PAIRS WITHOUT ODOMETRY. IN ADDITION, IT IS BY FAR THE FASTEST, ESPECIALLY WHEN EVIDENCE BASED SORTING IS USED.

Scan	ICP (Point-Point)				ICP (Point-Plane)				3D NDT				MUMC				
Pair	No Odo.		Odometry		No Odo.		Odometry		No Odo.		Odometry		No Odo.				
	100%		100%		100%		100%		100%		100%		100%	#	50%	#	
1,2	7.64	×	1.13	×	19.44	√	9.78	√	10.2	√	11.2	√	4.781	4	0.493	4	√
2,3	1.25	√	2.16	√	1.43	√	8.56	√	5.89	√	8.77	√	1.550	9	0.120	6	√
3,4	5.84	×	0.72	×	42.14	×	60.03	√	10.5	×	13.4	×	0.318	5	0.042	4	√
4,5	5.01	×	0.80	P	8.73	√	3.96	√	12.1	P	11.3	√	0.164	6	0.048	6	√
5,6	13.66	×	3.25	P	16.28	√	7.83	√	8.07	√	12.5	√	0.613	5	0.096	6	√



(a) The first sample r1_0 in the sequence.

(b) The last sample r1_20 in the sequence.

Fig. 7. Intensity images for the first and the last samples taken from [41]. The others can be viewed online.

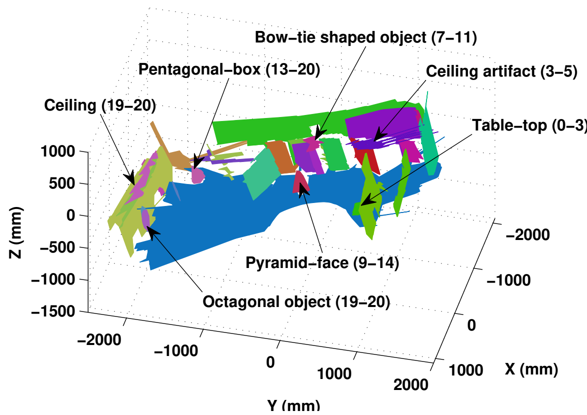


Fig. 8. Annotated front view with final fused planes. Note that even far away objects were matched and their geometry is recognizable. The number-ranges in parenthesis are the view numbers in the data-set in which the object is visible. Refer to website [41] for intensity images of the views. It appears that the LADAR was mounted on the robot tilted downwards.

with resolution 128×128, FOV $60^\circ \times 60^\circ$, and a relatively poor range-resolution of 3.66 cm. It is available online at [41] (left dataset, see Fig. 7). The data-set represents a sequence of samples taken from the LADAR mounted on a robot moving in a cluttered lab environment. No odometry or ground-truth is available. It has also been studied in [2, Fig. 17], who provide the result for the registration of only 9 samples (r1_1 to r1_9) out of 40.

The results of MUMC matching for the first 21 successive samples r1_0 to r1_20 are shown in Fig. 8 and 9. We have applied the coordinate-transform $\mathbf{x} \mapsto -\mathbf{x}, \mathbf{y} \mapsto \mathbf{z}, \mathbf{z} \mapsto \mathbf{y}$ to their point-clouds to retain our definition of roll-pitch-yaw angles. Figs. 9(a) and 9(b) show the computed robot pose change. MUMC reports insufficient overlap for the pair r1_20 to r1_21, which is also obvious to the human eye. Since no odometry is available, the sequence is broken. There are, however, other sequences which match.

Due to similar resolution, the time for region-growing and polygonization remain similar to that of the Swiss-Ranger. The number of planar patches per scan varied between 8 and 19. Since odometry was unavailable, we set $F_t = 100\%$, for which the MUMC matching time per pair varied between $0.1 - 0.66$ seconds.

D. Actuated LRF Data-sets

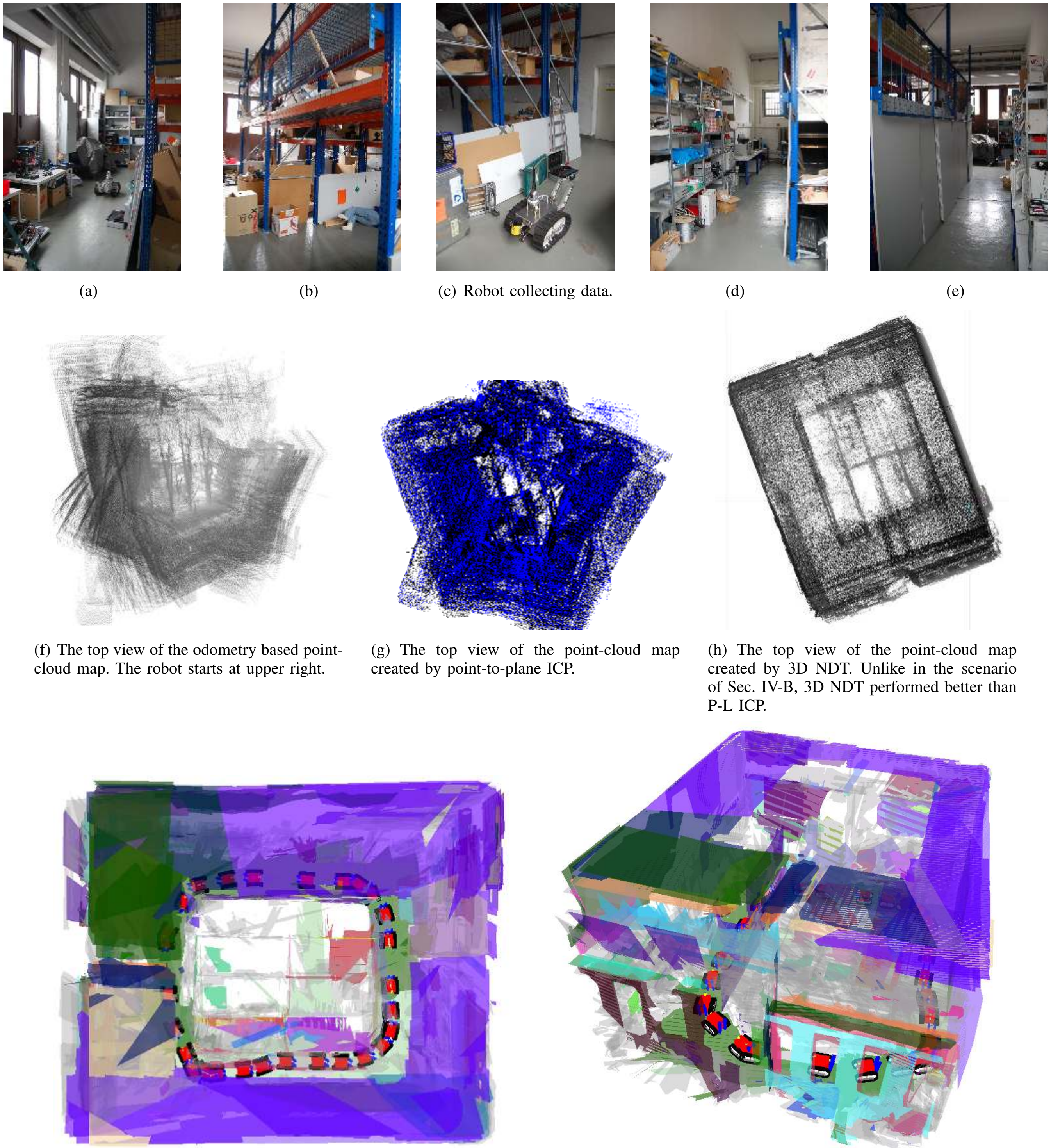
Finally, we provide some larger examples of data collected using a large FOV sensor, namely the ALRF. The ALRF provides only range-images but no intensity images. Two scenarios are considered.

1) Multistory Robot Rescue Arena: The robot was teleoperated to go around the arena, and data was collected in a stop-and-go fashion. A total of 29 scans were taken. Due to the discrete rotations of the actuator pitching the LRF, each scan took about 20 seconds. The main difficulty in this data-set are the large occlusions when the robot turns a corner.

The arena, its odometry map, its point-to-plane ICP map, and different views of its MUMC map are shown in Figs. 10 and 11. Odometry was provided to the algorithms. Point-to-plane ICP fully succeeded in 53.6% of pairs, partially succeeded in 35.7% of pairs, and failed for 10.7% of pairs. Point-to-point ICP was, as expected, less robust than point-to-plane ICP. It succeeded in only about 30% of pairs, partially succeeded in 35%, and failed in 35% of cases. 3D NDT performed much better than P-L ICP for this scenario than in the Swiss-Ranger scenario of Sec. IV-B. In both cases, its parameters were tuned by its designers [5].

The plane-fitting time per scan was on an average 3 seconds, and MUMC took about 0.8 seconds on average for matching a scan-pair for $F_t = 15\%$. Another 3 seconds are required for polygonizations of planar patches for visualization. The accuracy of the scan-matching can be seen by the fact that there is hardly any rotation or translation error present when the loop is closed and the robot returns near its starting location. Thus the map is already quite self-consistent even without any relaxation step.

2) Hannover Fair German Robocup 2009 Hall: Data was collected as before, except that the robot movements between scans were large: sometimes up to 5 meters in translation and 55° in rotation. No odometry was available. Due to



(a) (b) (c) Robot collecting data. (d) (e)

(f) The top view of the odometry based point-cloud map. The robot starts at upper right.

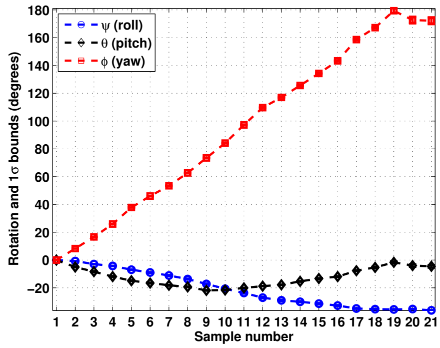
(g) The top view of the point-cloud map created by point-to-plane ICP.

(h) The top view of the point-cloud map created by 3D NDT. Unlike in the scenario of Sec. IV-B, 3D NDT performed better than P-L ICP.

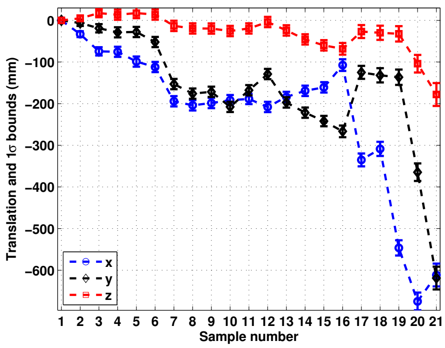
(i) The top view of the MUMC map. The robot starts at top left and goes clockwise around.

(j) The tilted top view showing the windows in the front which are visible in Fig. 10(a).

Fig. 10. The photos of the indoor multistory robot rescue arena are shown in Figs. 10(a) to 10(e). The robot goes one full round around it. The robot is shown at scan locations in the map– see also Fig. 11. The matched planes are shown in the same semi-transparent color, while unmatched planes are grayed out.



(a) Computed robot rotation. Sample 1 corresponds to r1_0, the zeroth view.



(b) Computed robot translation. Sample 1 corresponds to r1_0, the zeroth view.

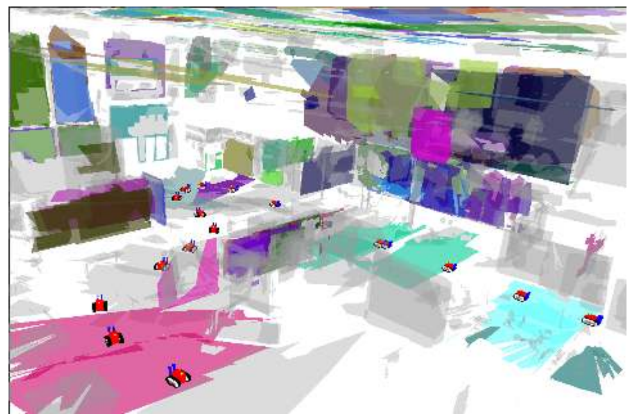
Fig. 9. The results from matching the first 21 views of the USF-Odetics r1 data-set.

Fig. 11. The inside view of the map of Fig. 10. The robot is shown at scan locations in the map by its 3D avatar which is true to scale– for a rough estimation it can be considered to be a cube of side 0.5 meter.

these constraints, more conservative size-similarity and overlap parameters had to be taken, viz. $\bar{L}_{\text{det}} = 7$ and $\bar{\chi}^2_{\text{ovlp}} = 1.25$, although $F_t = 20\%$ as before. The resulting map is shown in Fig. 12.

3) Other Data-sets and Multimedia: Further data-sets including a quite unstructured outdoor scenario at Disaster City are available at http://robotics.jacobs-university.de/projects/3Dmap/. Multimedia of all 3D maps are also provided there.

(a) View of the hall



(b) A tilted top view from below the ceiling.

Fig. 12. Hannover Fair German Robocup 2009 hall and its map created using MUMC. No odometry was provided to the algorithm. Note the relatively large area covered compared to the size of the robot.

V. Conclusions

The mathematical machinery for doing online pose registration based solely on planes extracted from range-images was presented. First, the simpler case of known plane correspondences was tackled. We derived expressions for least-squares pose and its covariance estimation considering plane-parameter uncertainty. Then, the work is extended to the general case by introducing a new algorithm for determining the unknown plane correspondences. This is done by maximizing geometric consistency. In doing so, a very efficient way to search the space of possible correspondences was introduced. To supplement the theoretical results, experiments for three 3D sensors were presented. Compared to ICP and 3D NDT, the presented algorithm is shown to be faster, to have a bigger convergence radius, and to require less memory.

Appendix A
Least Squares Plane Fitting

The planar patches are first extracted from a given range-image from a 3D sensor using a region-growing algorithm described in [29]. The uncertainty analysis of the plane extraction process is given in [17] and summarized here. Assume that a planar patch has been extracted, and is known to be composed of a set of points $\mathbf{p}_j$, $j = 1 \ldots N$. Their covariances $\mathbf{C}_{\mathbf{p},j}$, which are usually taken to be linear or quadratic functions of their respective $\|\mathbf{p}_j\|$, are also known. The weights and the weighted center of mass for these points is given by

$$w_j^{-1} \triangleq \text{trace}(\mathbf{C}_{\mathbf{p},j}), \tag{47}$$

$$\mathbf{p}_c \triangleq \frac{\sum_{j=1}^{\mathsf{N}} w_j \mathbf{p}_j}{\sum_{j=1}^{\mathsf{N}} w_j}, \qquad \mathbf{C}_{\mathbf{p}_c} = \Big(\sum_{j=1}^{\mathsf{N}} w_j\Big)^{-1}\mathbf{I}. \tag{48}$$

The very same weights have also been employed in [9]. Then the least-squares plane parameters are determined as follows.

$$\mathbf{M} \triangleq \sum_{j=1}^{\mathsf{N}} w_j(\mathbf{p}_j - \mathbf{p}_c)(\mathbf{p}_j - \mathbf{p}_c)^{\mathsf{T}}. \tag{49}$$

The plane-normal $\hat{\mathbf{n}}$ is the Eigenvector of $\mathbf{M}$ corresponding to its minimum Eigenvalue. The plane-parameter $d = \hat{\mathbf{n}}^{\mathsf{T}}\mathbf{p}_c$, as the plane passes through $\mathbf{p}_c$. The covariance of the plane-parameters is $\mathbf{C} = -\mathbf{H}^{+}$, where the Hessian matrix $\mathbf{H}$ is defined as

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{\hat{\mathbf{n}}\hat{\mathbf{n}}} & \mathbf{H}_{\hat{\mathbf{n}}d} \\ \mathbf{H}_{\hat{\mathbf{n}}d}^{\mathsf{T}} & \mathbf{H}_{dd} \end{bmatrix}, \text{ where,} \tag{50a}$$

$$\mathbf{H}_{dd} = -\sum_{j=1}^{\mathsf{N}} w_j, \quad \mathbf{H}_{\hat{\mathbf{n}}d} = -\mathbf{H}_{dd}\mathbf{p}_c, \tag{50b}$$

$$\mathbf{H}_{\hat{\mathbf{n}}\hat{\mathbf{n}}} = -\mathbf{M} + \mathbf{H}_{dd}\mathbf{p}_c\mathbf{p}_c^{\mathsf{T}} + (\hat{\mathbf{n}}^{\mathsf{T}}\mathbf{M}\hat{\mathbf{n}})\mathbf{I}_3. \tag{50c}$$

From these expressions, it can be derived that

$$\mathbf{H}\begin{bmatrix}\hat{\mathbf{n}} \\ d\end{bmatrix} = \mathbf{0} \qquad \Rightarrow \mathbf{C}\begin{bmatrix}\hat{\mathbf{n}} \\ d\end{bmatrix} = \mathbf{0}. \tag{51}$$

Appendix B
Decoupling the Covariances

To be able to use the nicely decoupled equations (4) for determining rotation and translation separately, we need to estimate the total uncertainty in $\hat{\mathbf{n}}$ by marginalizing, i.e. integrating out the effect of d and vice-versa. Standard results for marginalization of Gaussians are not directly applicable because they are written in terms of the covariance matrix and not the Hessian; the latter is what is directly estimated [17], and the former is computed from it by Moore-Penrose inverse. We show that the marginalization can still be easily performed, if we start from first principles.

Let $\bar{\boldsymbol{\nu}} \triangleq (\bar{\mathbf{n}}^{\mathsf{T}}, \bar{d})^{\mathsf{T}}$ be the mean plane parameters and we define the perturbations about the mean as

$$\Delta\boldsymbol{\nu} \triangleq \begin{bmatrix}\Delta\hat{\mathbf{n}} \\ \Delta d\end{bmatrix} \qquad \triangleq \begin{bmatrix}\hat{\mathbf{n}} - \bar{\mathbf{n}} \\ d - \bar{d}\end{bmatrix}. \tag{52}$$

The corresponding joint probability distribution function (PDF) is given in terms of the plane's Hessian $\mathbf{H}$ as

$$p(\boldsymbol{\nu} \mid \bar{\boldsymbol{\nu}}, \mathbf{H}) = \eta \exp\left\{\frac{1}{2}\Delta\boldsymbol{\nu}^{\mathsf{T}}\begin{bmatrix} \mathbf{H}_{\hat{\mathbf{n}}\hat{\mathbf{n}}} & \mathbf{H}_{\hat{\mathbf{n}}d} \\ \mathbf{H}_{\hat{\mathbf{n}}d}^{\mathsf{T}} & \mathbf{H}_{dd} \end{bmatrix}\Delta\boldsymbol{\nu}\right\}, \tag{53}$$

where, η is the normalizing constant, and the Hessian has been written in a partitioned form in the exponent. This PDF is defined on the tangent-plane of the domain of $\boldsymbol{\nu}$ at $\bar{\boldsymbol{\nu}}$, which implies that

$$(\Delta\hat{\mathbf{n}}) \cdot \bar{\mathbf{n}} \approx 0. \tag{54}$$

From the first of (51) it can be deduced that

$$\mathbf{H}_{dd} - \mathbf{H}_{\hat{\mathbf{n}}d}^{\mathsf{T}}\mathbf{H}_{\hat{\mathbf{n}}\hat{\mathbf{n}}}^{-1}\mathbf{H}_{\hat{\mathbf{n}}d} = 0. \tag{55}$$

Using this we can algebraically decompose the exponent of the joint PDF by the method of completion of squares as

$$\Delta\boldsymbol{\nu}^{\mathsf{T}}\mathbf{H}\Delta\boldsymbol{\nu} = \boldsymbol{\xi}^{\mathsf{T}}\mathbf{H}_{\hat{\mathbf{n}}\hat{\mathbf{n}}}\boldsymbol{\xi}, \quad \boldsymbol{\xi} \triangleq \Delta\hat{\mathbf{n}} + \Delta d\,\mathbf{H}_{\hat{\mathbf{n}}\hat{\mathbf{n}}}^{-1}\mathbf{H}_{\hat{\mathbf{n}}d}, \tag{56}$$

which shows that the random variable $\boldsymbol{\xi} \in \mathbb{R}^3$ is normally distributed with mean $\mathbf{0}$ and covariance $-\mathbf{H}_{\hat{\mathbf{n}}\hat{\mathbf{n}}}^{-1}$. Taking the component of $\boldsymbol{\xi}$ along $\bar{\mathbf{n}}$ and using (54) gives

$$\bar{\mathbf{n}}^{\mathsf{T}}\boldsymbol{\xi} = \Delta d\,\bar{\mathbf{n}}^{\mathsf{T}}\mathbf{H}_{\hat{\mathbf{n}}\hat{\mathbf{n}}}^{-1}\mathbf{H}_{\hat{\mathbf{n}}d}. \tag{57}$$

This finally allows us to derive the decoupled covariance of Δd as

$$\mathbf{D}_{dd} = \frac{-\bar{\mathbf{n}}^{\mathsf{T}}\mathbf{H}_{\hat{\mathbf{n}}\hat{\mathbf{n}}}^{-1}\bar{\mathbf{n}}}{\left(\bar{\mathbf{n}}^{\mathsf{T}}\mathbf{H}_{\hat{\mathbf{n}}\hat{\mathbf{n}}}^{-1}\mathbf{H}_{\hat{\mathbf{n}}d}\right)^2}. \tag{58}$$

To compute the decoupled covariance of $\Delta\hat{\mathbf{n}}$, we decompose the exponent of the joint PDF again by method of completion of squares as

$$\begin{aligned}\Delta\boldsymbol{\nu}^{\mathsf{T}}\mathbf{H}\Delta\boldsymbol{\nu} &= \Delta\hat{\mathbf{n}}^{\mathsf{T}}(\mathbf{H}_{\hat{\mathbf{n}}\hat{\mathbf{n}}} - \mathbf{H}_{\hat{\mathbf{n}}d}\mathbf{H}_{dd}^{-1}\mathbf{H}_{\hat{\mathbf{n}}d}^{\mathsf{T}})\Delta\hat{\mathbf{n}} \\ &\quad + \mathbf{H}_{dd}\left(\Delta d + \mathbf{H}_{dd}^{-1}\mathbf{H}_{\hat{\mathbf{n}}d}^{\mathsf{T}}\Delta\hat{\mathbf{n}}\right)^2.\end{aligned} \tag{59}$$

This decomposition can be used to integrate out Δd from the joint PDF, and we get the decoupled Hessian for $\Delta\hat{\mathbf{n}}$ as

$$\mathbf{H}'_{\hat{\mathbf{n}}\hat{\mathbf{n}}} \triangleq \mathbf{H}_{\hat{\mathbf{n}}\hat{\mathbf{n}}} - \mathbf{H}_{\hat{\mathbf{n}}d}\mathbf{H}_{dd}^{-1}\mathbf{H}_{\hat{\mathbf{n}}d}^{\mathsf{T}}. \tag{60}$$

Using (51) it can be verified that we have the nice property $\mathbf{H}'_{\hat{\mathbf{n}}\hat{\mathbf{n}}}\bar{\mathbf{n}} = 0$. Finally, the decoupled covariance of $\Delta\hat{\mathbf{n}}$ is

$$\mathbf{D}_{\hat{\mathbf{n}}\hat{\mathbf{n}}} \triangleq -(\mathbf{H}'_{\hat{\mathbf{n}}\hat{\mathbf{n}}})^{+}, \qquad \mathbf{D}_{\hat{\mathbf{n}}\hat{\mathbf{n}}}\bar{\mathbf{n}} = \mathbf{0}. \tag{61}$$

References

[1] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robotics and Automation Magazine*, pp. 99–108, June 2006.

[2] D. Fischer and P. Kohlhepp, "3D geometry reconstruction from multiple segmented surface descriptions using neuro-fuzzy similarity measures," *Journal of Intelligent and Robotic Systems*, vol. 29, pp. 389–431, 2000.

[3] H. Surmann, A. Nuechter, and J. Hertzberg, "An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments," *Robotics and Autonomous Systems*, vol. 45, no. 3-4, pp. 181–198, 2003.

[4] J. Weingarten and R. Siegwart, "3D SLAM using planar segments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Beijing, 2006.

[5] M. Magnusson, A. Lilienthal, and T. Duckett, "Scan registration for autonomous mining vehicles using 3D-NDT," *Journal of Field Robotics*, vol. 24, no. 10, pp. 803–827, 2007.

[6] A. Nüchter, K. Lingemann, and J. Hertzberg, "6D SLAM– 3D mapping outdoor environments," *Journal of Field Robotics*, vol. 24, no. 8/9, pp. 699–722, 2007.

[7] CSEM, *The SwissRanger, Manual V1.02*, 8048 Zurich, Switzerland, 2006. [Online]. Available: http://www.swissranger.ch

[8] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, Feb 1992.

[9] J. Weingarten, "Feature-based 3D SLAM," Ph.D. dissertation, EPFL, Lausanne, Switzerland, 2006. [Online]. Available: http://library.epfl.ch/theses/?nr=3601

[10] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," in *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*, 1991, pp. 2724–2729 vol.3.

[11] ——, "Object Modeling by Registration of Multiple Range Images," *Image and Vision Computing*, vol. 10, no. 3, pp. 145–155, 1992.

[12] M. W. M. G. Dissanayake, P. Newman, S. Clark, and H. F. Durrant-Whyte, “A solution to the simultaneous localization and map building (SLAM) problem,” *IEEE Trans. on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, June 2001.

[13] B. K. P. Horn and J. G. Harris, “Rigid body motion from range image sequences,” *CVGIP: image understanding*, vol. 53, pp. 1–13, 1991.

[14] H. Gharavi and S. Gao, “3-D motion estimation using range data,” *IEEE Trans. on Intelligent Transportation Systems*, vol. 8, pp. 133–143, 2007.

[15] O. D. Faugeras and F. Lustman, “Motion and structure from motion in piecewise planar environment,” *Int. Journal of Pattern Recognition and Artificial Intelligence*, vol. 2, no. 3, pp. 485–508, 1988.

[16] K. Pathak, A. Birk, N. Vaskevicius, M. Pfingsthorn, S. Schwertfeger, and J. Poppinga, “Online 3D SLAM by Registration of Large Planar Surface Segments and Closed Form Pose-Graph Relaxation,” *Journal of Field Robotics, Special Issue on 3D Mapping*, vol. 27, no. 1, pp. 52–84, 2010.

[17] K. Pathak, N. Vaskevicius, and A. Birk, “Revisiting Uncertainty Analysis for Optimum Planes Extracted from 3D Range Sensor Point-Clouds,” in *International Conference on Robotics and Automation (ICRA)*. IEEE press, 2009, pp. 1631 – 1636.

[18] ——, “Uncertainty Analysis for Optimum Plane Extraction from Noisy 3D Range-Sensor Point-Clouds,” *Intelligent Service Robotics*, vol. 3, pp. 37–48, 2010.

[19] P. Kohlhepp, P. Pozzo, M. Walther, and R. Dillmann, “Sequential 3D-SLAM for mobile action planning,” *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 1, pp. 722–729 vol.1, Sept.-2 Oct. 2004.

[20] P. Kohlhepp, G. Bretthauer, M. Walther, and R. Dillmann, “Using orthogonal surface directions for autonomous 3d-exploration of indoor environments,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2006, pp. 3086–3092.

[21] R. Fisher, “Geometric constraints from planar surface patch matching,” *Image and Vision Computing*, vol. 8, no. 2, pp. 148–154, 1990.

[22] W. E. L. Grimson, *Object recognition by computer: the role of geometric constraints*. Cambridge: MIT Press, 1990.

[23] W. He, W. Ma, and H. Zha, “Automatic registration of range images based on correspondence of complete plane patches,” in *Fifth Int. Conf. on 3-D Digital Imaging and Modeling (3DIM’05)*. IEEE, 2005, pp. 470–475.

[24] D. Viejo and M. Cazorla, “3D plane-based egomotion for SLAM on semi-structured environment,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2007, pp. 2761–2766.

[25] A. Makadia, A. Patterson, and K. Daniilidis, “Fully automatic registration of 3D point clouds,” in *IEEE Comp. Soc. Conf. on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, June 2006, pp. 1297–1304.

[26] M. A. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Graphics and Image Processing*, vol. 24, no. 6, pp. 381–395, 1981.

[27] D. Fontanneli, L. Ricciato, and S. Soatto, “A fast RANSAC-based registration algorithm for accurate localization in unknown environments using LIDAR measurements,” in *IEEE Int. Conf. on Automation Science and Engineering (CASE)*, 2007, pp. 597–602.

[28] J. J. Craig, *Introduction to robotics – Mechanics and control*. Prentice Hall, 2005.

[29] J. Poppinga, N. Vaskevicius, A. Birk, and K. Pathak, “Fast plane detection and polygonalization in noisy 3D range images,” in *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, Nice, France, 2008.

[30] ——, “Fast Plane Detection and Polygonalization in noisy 3D Range Images,” in *International Conference on Intelligent Robots and Systems (IROS)*. Nice, France: IEEE Press, 2008, pp. 3378 – 3383.

[31] Y. Kanazawa and K. Kanatani, “Reliability of fitting a plane to range data,” *IEICE Transactions on Information and Systems*, vol. E78-D, no. 12, pp. 1630–1635, 1995.

[32] K. Kanatani, *Statistical Optimization for Geometric Computation*. New York: Dover Publications, Inc., 2005, iSBN 0486443086.

[33] D. S. Sivia, *Data Analysis: A Bayesian Tutorial*. Oxford University Press, 1996.

[34] B. K. P. Horn, “Closed-form solution of absolute orientation using unit quaternions,” *Journal of the Optical Society of America A*, vol. 4, pp. 629–642, 1987.

[35] M. D. Shuster, “The generalized Wahba problem,” *The Journal of the Astronautical Sciences*, vol. 54, no. 2, pp. 245–259, April-June 2006.

[36] F. L. Markley, “Attitude estimation or quaternion estimation?” in *Flight Mechanics Symposium*. NASA Goddard Space Flight Center, 2003.

[37] A. Carmi and Y. Oshman, “On the covariance singularity of quaternion estimators,” in *AIAA Guidance, Navigation and Control Conference and Exhibit*, South Carolina, 2007.

[38] G. H. Golub and C. F. V. Loan, “An analysis of the total least squares problem,” *SIAM Journal of Numerical Analysis*, vol. 17, no. 6, pp. 883–893, 1980.

[39] D. M. Mount and S. Arya, *ANN: A Library for Approximate Nearest Neighbor Searching*, University of Maryland, 2006. [Online]. Available: http://www.cs.umd.edu/∼mount/ANN/

[40] M. Magnusson, A. Nüchter, C. Lörken, A. J. Lilienthal, and J. Hertzberg, “Evaluation of 3d registration reliability and speed: A comparison of ICP and NDT,” in *IEEE International Conference on Robotics and Automation*, Kobe, Japan, 2009.

[41] A. Hoover, “Ego-motion sequence using Odetics LADAR r1, Image Analysis Research Laboratory, Univ. South Florida,” 1994. [Online]. Available: http://marathon.csee.usf.edu/range/icons/Odet-cam-mot.html

Kaustubh Pathak received his Ph.D. in Mechanical Engineering in Fall 2005 from the University of Delaware, Newark, USA, in the area of nonlinear control of underactuated and nonholonomic systems. Since 2006, he is a post-doctoral fellow with the robotics group of the department of EECS at the Jacobs University Bremen, Germany. His research interests include probabilistic feature extraction and fusion, 3D mapping, path-planning, and control of mobile robots. He has authored over 35 papers in international journals and conferences.

Andreas Birk is an associate professor in Electrical Engineering and Computer Science (EECS) at Jacobs University Bremen. He started at Jacobs University in Fall 2001, building up the robotics group at this newly founded private university. Before he joined Jacobs, he held a research-mandate from the Flemish Society for Applied Research (IWT) and was a visiting Professor at the Vrije Universiteit Brussel (VUB) where he did his research in the Artificial Intelligence Lab (AI-lab). In 1995 he received his doctorate from the Universität des Saarlandes, Saarbrücken. Andreas Birk’s research focuses on autonomous systems. On the engineering side, he is working on the design and construction of complete systems. This includes the design and construction of embedded hardware and mechatronics as well as software development up to full autonomy. On the basic research side, he is interested in a constructive understanding of intelligence.

Narūnas Vaškevičius received his MSc. degree in computer science with robotics specialization from Jacobs University Bremen in 2008. He is currently pursuing a Ph.D degree in computer science as a member of Jacobs Robotics Lab. His research concentrates on autonomous robotics in the context of path-planning and 3D mapping based on probabilistic surface feature extraction. Among others, he applies his research in the fields of space and underwater robotics, as well as SSRR.

Jann Poppinga received a Diplom (M.Sc. equivalent) in Computer Science from the University of Oldenburg in 2006 under the supervision of Prof. Appelrath. He continued his studies at PhD level at Jacobs University under supervision of Prof. Birk. His research interests are 3D data processing, especially 3D path planning and mapping for mobile robots. He has authored 4 publications.