Automated Three-Dimensional Axis Mapping with a Mobile Platform*

Marc J. Gallant and Joshua A. Marshall

Abstract—An axis map (AM) represents the orientations of planar surfaces in an environment and is void of positional information. Three-dimensional axis mapping (3DAM) is a graph-based optimization algorithm that generates an AM, while carefully addressing the parameterization of axes (i.e., normal vectors). 3DAM exploits the lack of positional information to form a densely connected graph. This paper provides a detailed description of the parameterization of axes, an outline of the 3DAM algorithm, as well as outdoor experimental results showing the usefulness of AMs as a tool for mobile autonomous geotechnical mapping.

I. INTRODUCTION

Every good mapping algorithm should employ a map representation that facilitates estimation, while at the same time remaining adequate for its application. For instance, representing an indoor environment as a set of perpendicular planes can significantly improve robotic mapping accuracy [1]. In the case of three-dimensional axis mapping (3DAM), axis maps (AMs) fulfill this purpose. An AM represents the orientations of planar surfaces in an environment by using a special parameterization of normal vectors called axes. AMs, like many topological maps, are void of positional information and lack scale in the Cartesian sense. However, like metric maps, geometrical information (i.e., the axis of a planar surface) is important to the representation. Unlike both topological and metric maps, the correlation between the planar surfaces in the environment and the entries of an AM is surjective. In other words, multiple physically distinct surfaces in the environment may be represented by a single entry of an AM.

3DAM is designed to estimate AMs while carefully addressing its parameterization and taking advantage of its surjectivity. Observations of the orientation of a mobile platform (e.g., a mobile robot, UAV, or handheld unit) using an IMU are combined with observations of an AM to form a graph optimization problem in the domain of rotations and axes. These observations are illustrated in Figure 1.

One major application area for AMs is as a tool for geologists and geotechnical engineers to help analyze the stability and formation of rock masses. Many engineering projects (e.g., the construction of roads, buildings, bridges, mines, etc.) require interaction with rock masses, which have highly anisotropic properties due to the existence of discontinuities (e.g., fractures). When part of a rock mass is exposed (called a rock face), discontinuities manifest themselves as a small number (typically < 5) of sets of planar surfaces (called joint sets), each statistically distributed in orientation (e.g., see in Figure 2). Measurements of these planar surfaces are usually performed by using hand tools and could be made much safer, more efficient, and less biased by using 3DAM.

A. Related Work

Using planar surfaces in the context of simultaneous localization and mapping (SLAM) is not new. Weingarten and Siegwart [2] used polygons comprised of planar segments as features in EKF-SLAM to significantly reduce the size of the map. Nguyen et al. [1] built extremely accurate indoor maps comprised of planar segments by taking advantage of an orthogonality assumption of planes common in indoor areas. Pathak et al. [3] created 3D-Plane-SLAM and highlighted the robustness and accuracy of plane registration with respect to rotations. The resulting rotational accuracy is argued to justify optimization of only the translation components of the constructed pose-graph. These methods are concerned with the position and orientation of distinct planes in the environment and depend on plane-to-plane or point-to-plane matching using LiDAR measurements. This is in contrast

*This work was funded in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) under project 397985-2011 and by the NSERC Canadian Field Robotics Network (NCFRN).

M. J. Gallant and J. A. Marshall are with the Mining Systems Laboratory, Faculty of Engineering and Applied Science, Queen’s University, Kingston, ON, Canada K7L 3N6 {m.gallant,joshua.marshall}@queensu.ca, https://msl.engineering.queensu.ca.
Several parameterizations of plane orientations have been used for state estimation. Lee et al. [4] used spherical coordinates, which is a minimal representation (i.e., two dimensions), making it convenient to represent as a random variable, but is subject to singularities. This was addressed by Kaess [5] by using projective geometry and the homogeneous representation of planes. Here, a full description of a plane is shown to be representable as a point in $S^3$, allowing parameterization as a unit quaternion for optimization. Finally, Hertzberg et al. [6] describes a general approach to parameterizing many different geometries for state estimation, including points in $S^3$ (which can describe the normal to a plane). Our approach follows that of Kaess and Hertzberg, where a global parameterization (unit axes) and minimal representation (axis vectors) are used simultaneously.

Some efforts have been made to map joint sets using LiDAR [7], [8], [9]. These approaches use stationary, tripod-mounted LiDAR where rock faces are scanned and planar surfaces are extracted from the resulting point clouds. This method has several disadvantages: the LiDAR systems are prohibitively large and expensive for some deployments, several scans from different perspectives are required to compensate for occlusions, closed-in areas (e.g., underground mines) restrict the position of the LiDAR, and mapping even small rock faces can be time consuming (e.g., many hours [8]). In contrast, 3DAM uses compact and comparatively inexpensive sensors, its mobility makes capturing all joint sets simpler (even in closed-in areas), and data collection is extremely fast. Finally, because 3DAM permits mobile data collection, remotely operated robotic or autonomous vehicles can safely map joint sets in unstable or unreachable places.

B. About This Paper

Section II discusses preliminary material including the parameterization of axes. Section III formulates the 3DAM algorithm itself. Finally, in Section IV, experimental results using 3DAM are presented. The results focus on the geotechnical application for 3DAM discussed in Section I.

II. PRELIMINARIES AND NOTATION

A. Rotations

Many different parameterizations of rotations have been explored in the context of state estimation [10]. One widely used parameterization is the unit quaternion $q$, defined as the $4 \times 1$ column

$$ q := \begin{bmatrix} \varepsilon \\ \eta \end{bmatrix}, $$

where $\varepsilon \in \mathbb{R}^3$ is the vector part and $\eta \in \mathbb{R}$ is the scalar part of the unit quaternion, and

$$ \|q\| = 1, \quad q \equiv -q, \quad q^{-1} = \begin{bmatrix} -\varepsilon \\ \eta \end{bmatrix}. $$

A rotation by $\theta \in \mathbb{R}$ about $a \in S^2$ is parameterized by $q$ as

$$ q = \begin{bmatrix} \sin \left( \frac{\theta}{2} \right) a \\ \cos \left( \frac{\theta}{2} \right) \end{bmatrix}, $$

Note that $q$ and $-q$ represent the same rotation; therefore, it is convenient to maintain $\eta \geq 0$. The null rotation ($\theta = 0$) results in the identity unit quaternion $q := [0^\top \ 1]^\top$.

The composition of two rotations parameterized by unit quaternions $p$ and $q$ is achieved using the quaternion product $p \otimes q$, where

$$ p \otimes q := \begin{bmatrix} \eta_p \varepsilon_q + \eta_q \varepsilon_p - \varepsilon^\times \varepsilon_q \\ \eta_p \eta_q - \varepsilon^\times \varepsilon_p \end{bmatrix}, $$

and $\varepsilon^\times$ is the usual skew-symmetric cross-product matrix of a 3-vector. Factoring $p$ or $q$ out of (4) transforms $p \otimes q$ into $p^\oplus q$ or $q^\oplus p$, where

$$ p^\oplus = \begin{bmatrix} \eta_p I_{3 \times 3} - \varepsilon^\times \\ -\varepsilon^\times \eta_p \end{bmatrix}, \quad q^\oplus = \begin{bmatrix} \eta_q I_{3 \times 3} + \varepsilon^\times \varepsilon_q \\ -\varepsilon^\times \eta_q \end{bmatrix}, $$

are the left-hand ($\oplus$) and right-hand ($\oplus$) compound operators on a unit quaternion. Several additional identities using this notation are described by Barfoot et al. [11].

Let $q_{ij}$ be a unit quaternion parameterizing a rotation from coordinate frame $i$ to coordinate frame $j$, and let $x_i \in \mathbb{R}^3$ be a vector expressed in coordinate frame $i$. Then

$$ x_j = q_{ij}^+ x_i q_{ij}^{-1}, $$

or more compactly, $x_j = C(q_{ij}) x_i$, where

$$ q_{ij}^+ q_{ji}^{-1} = \begin{bmatrix} C(q_{ij}) & 0 \\ 0 \ 1 \end{bmatrix}. $$

Parameterizing a rotation as a random variable (i.e., a random rotation) with a unit quaternion is troublesome because it represents three degrees of freedom with four parameters and the unit constraint. This is usually addressed by representing rotation error by a minimal representation, where—given some assumptions about the operating conditions—the singularities associated with these representations are avoided [12]. We use a rotation vector to represent the error associated with a unit quaternion. Given a rotation error of angle $\delta \theta \in \mathbb{R}$ about axis $a \in S^2$, the corresponding rotation vector is defined as $\delta \theta := \delta \theta a$. As a result, a random rotation parameterized by a unit quaternion has a $3 \times 3$ covariance matrix given by $E\left[ (\delta \theta) \delta \theta^\top \right]$.

B. Axes

An axis is the unordered pair of directions diametrically opposed on the unit sphere in $\mathbb{R}^3$, or equivalently, a single point on the unit hemisphere. Axes have two degrees of freedom and, like rotations, minimal parameterizations of axes have at least one singularity. Axes can be used to represent the orientation of a plane in $\mathbb{R}^3$, as illustrated in Fig. 3. A unit axis $n$ is the $3 \times 1$ column

$$ n := \begin{bmatrix} \kappa \\ \lambda \end{bmatrix}, $$

where $\kappa \in \mathbb{R}^2$ is the vector part and $\lambda \in \mathbb{R}$ is the scalar part of the unit axis, and

$$\|\mathbf{n}\| = 1, \quad \mathbf{n} \equiv -\mathbf{n}, \quad \mathbf{n}^{-1} = \begin{bmatrix} -\kappa \\ \lambda \end{bmatrix}.$$  

(9)

A unit axis $\mathbf{n}$ and its components are illustrated in Fig. 4, including its inclination $\phi \in \left[0, \frac{\pi}{2}\right]$ (the angle between $\mathbf{n}$ and the axis of the hemisphere) and its direction $\mathbf{r} \in S^1$ (the normalized vector part). An axis with inclination $\phi$ and declination $\mathbf{r}$ is parameterized by $\mathbf{n}$ as

$$\mathbf{n} = \begin{bmatrix} \sin(\phi) \mathbf{r} \\ \cos(\phi) \end{bmatrix}.$$  

(10)

Note that $\mathbf{n}$ and $-\mathbf{n}$ represent the same axis; therefore, it is convenient to maintain $\lambda \geq 0$. The null axis ($\phi = 0$) results in the identity unit axis $\mathbf{o} := [1, 0]$. Note the similarities between unit quaternions and unit axes in terms of their properties and identity elements.

The composition of two axes parameterized by unit axes $\mathbf{n}$ and $\mathbf{m}$ is achieved using the axis product $\mathbf{n} \otimes \mathbf{m}$, where

$$\mathbf{n} \otimes \mathbf{m} := \mathbf{n}^+ \mathbf{m},$$  

(11)

and $\mathbf{n}^+$ is the rotation matrix in that brings $\mathbf{o}$ to $\mathbf{n}$; i.e.,

$$\mathbf{n}^+ = \lambda_n I_3 + \frac{1 - \lambda_n}{\|\kappa_n\|^2} \kappa_n \kappa_n^T + \kappa_n \kappa_n^T,$$  

(12)

where

$$\mathbf{n} = \begin{bmatrix} \kappa_n \\ \lambda_n \end{bmatrix}, \quad \kappa_n = \begin{bmatrix} \kappa_1 \\ \kappa_2 \end{bmatrix}, \quad \kappa_n = \begin{bmatrix} -\kappa_2 & \kappa_1 \end{bmatrix}.$$  

(13)

The rotation parameterized by $\mathbf{n}^+$ can be derived geometrically as a rotation of $\mathbf{o}$ by the inclination of $\mathbf{n}$ about $\mathbf{a} \in S^2$, where $\mathbf{a}$ is perpendicular to both $\mathbf{o}$ and the direction of $\mathbf{n}$.

Like random rotations, the error associated with a random axis is parameterized by using a minimal representation. We use an axis vector to represent the error associated with a unit axis. Given an error with inclination $\delta \phi$ in the direction $\mathbf{r}$, the corresponding axis vector is defined as $\delta \mathbf{a} := \delta \phi \mathbf{a}$. As a result, a random axis parameterized by a unit axis has a $2 \times 2$ covariance matrix given by $E[(\delta \phi) (\delta \phi)^T]$.

**C. Manifold Encapsulation**

The constrained parameterizations of rotations (unit quaternions) and axes (unit axes) are global representations, free of singularities. However, most state estimation algorithms operate on the local neighbourhood of the state and observations as if they are random vectors in $\mathbb{R}^n$. Fortunately, the local neighbourhoods relative to unit quaternions and unit axes can be mapped by a homeomorphism to the domains of rotation vectors and axis vectors, respectively, which behave like vector spaces in these neighbourhoods. A detailed investigation of this approach for many different manifolds was performed by Hertzberg et al. [6].

We define two operators that provide transformations between the global $(\mathcal{M})$ and local $(\mathcal{R}^n)$ parameterizations,

$$\Box : \mathbb{M} \times \mathbb{R}^n \rightarrow \mathbb{M},$$  

$$\Box : \mathbb{M} \times \mathbb{M} \rightarrow \mathbb{R}^n.$$  

(14)

Given unit quaternions $\mathbf{p}$ and $\mathbf{q}$, and a rotation vector $\theta$ that is in the local neighbourhood of $\mathbf{q}$,

$$\mathbf{q} \boxplus \theta := \mathbf{q}^+ \exp \left( \frac{\theta}{2} \right), \quad \mathbf{q} \boxminus \mathbf{p} := 2 \log \left( \mathbf{p}^{-1} + \mathbf{q} \right),$$  

(15)

where

$$\exp(\theta) := \begin{bmatrix} \sin(\theta) \theta \\ \cos(\theta) \end{bmatrix},$$  

$$\log(\mathbf{q}) := \begin{cases} \frac{\theta}{2} & \text{for } \mathbf{q} = \mathbf{t}, \\ \frac{\theta}{2} \varepsilon & \text{for } \eta = 0, \\ \frac{\theta}{\sqrt{\eta^2 + \varepsilon^2}} \varepsilon & \text{otherwise.} \end{cases}$$  

(16)

Similarly, given unit axes $\mathbf{m}$ and $\mathbf{n}$, and an axis vector $\phi$ that is in the local neighbourhood of $\mathbf{n}$,

$$\mathbf{n} \boxplus \phi := \mathbf{n}^+ \exp(\phi), \quad \mathbf{m} \boxminus \mathbf{n} := \log(\mathbf{m}^{-1} + \mathbf{n}),$$  

(17)

where

$$\exp(\phi) := \begin{bmatrix} \sin(\phi) \phi \\ \cos(\phi) \end{bmatrix},$$  

$$\log(\mathbf{n}) := \begin{cases} \frac{\phi}{\sqrt{\kappa^2 + \lambda^2}} \kappa & \text{for } \kappa = 0, \\ \frac{\phi}{\tan^{-1}(\|\kappa\|, \lambda)} \kappa & \text{otherwise.} \end{cases}$$  

(18)

In (16) and (18), the the unnormalized sinc function has the property sinc(0) := 0.

**D. Definitions**

The orientation path $\mathbf{Q}$ is a discrete sequence of the orientations of a mobile platform as it moves through an environment, expressed as rotations from the global coordinate frame $\mathcal{F}_g$ (North-East-Down) to the mobile platform coordinate frame $\mathcal{F}_m$. The axis map $\mathbf{A}$ represents the axes.
of surfaces in an environment expressed in $F_g$. In 3DAM, the orientation path and axis map are

$$Q = (q_1, q_2, \ldots, q_n),$$

$$A = (n_1, n_2, \ldots, n_m),$$

where $q_i \in Q$ are parameterized as unit quaternions, and $n_j \in A$ are parameterized as unit axes. Let $Q$ be an estimate of $Q$, where the entries of $Q$ are random rotations. Similarly, let $A$ be an estimate of $A$, where the entries of $A$ are random axes. Let $X = (Q, A)$, where $X$ is an estimate of $X$ with a $(3n+2m) \times (3n+2m)$ covariance matrix $P$ (recall that errors of unit quaternions and unit axes are parameterized by minimal representations).

III. THREE-DIMENSIONAL AXIS MAPPING

Three dimensional axis mapping (3DAM) is an algorithm that estimates the axes of the planar surfaces in an environment. Data is collected by a mobile platform (MP) that is equipped with an inertial measurement unit (IMU) and a three-dimensional LiDAR. The MP is moved through an environment while scanning the surfaces to be mapped. A high-level overview of the 3DAM algorithm is provided in Fig. 5 and is summarized by the following steps.

1) Initialize $Q$ using the IMU data in a Kalman Filter (KF) (Section III-A).
2) Generate rotation observations with $Q$ (Section III-B).
3) Generate axis observations from the LiDAR measurements. Estimate $A$ by clustering all axis observations in $F_g$. Associate axis observations with entries of $A$ or mark them as outliers (Section III-C).
4) Join $Q$ and $A$ to form a state estimate $\hat{X}$. Using $\hat{X}$, the observations, and the manifold operations defined in Section II-C, solve for the state perturbation $\Delta \hat{X}$ that minimizes the negative log-likelihood of the observations and apply it to $\hat{X}$ (Section III-D). If convergence criteria are not met, return to Step 3. Otherwise, rotate all axis observations to $F_g$ using the optimal OP estimate $\hat{Q}^*$ to produce the final AM. The following sections provide more detailed descriptions of each of these steps.

A. Kalman Filter

A Kalman Filter (KF) based on a similar filter by Trawny [13] is used to provide an initial estimate of $Q$. The KF uses the measurements of an inertial measurement unit (IMU) containing a three-axis accelerometer, three-axis gyroscope, and three-axis magnetometer. The first orientation $\hat{q}_1$ is estimated by using the factored quaternion algorithm [14]. $A$ priori estimates are calculated by using the gyroscope measurements and a zero-th order quaternion integrator (i.e., the angular velocity is assumed constant over the short interval between measurements). $A$ posteriori estimates are calculated by using the measured direction of gravity and the Earth’s magnetic field by the accelerometer and magnetometer, respectively, and their corresponding known directions in $F_g$. The uncertainty of the measured direction of gravity is inflated if external accelerations are detected. The KF estimates at each LiDAR measurement are taken to form the estimated orientation path $\hat{Q}$.

B. Rotation Observations

Because the accelerometer and magnetometer used by the KF in Section III-A measure an absolute reference (i.e., the direction of gravity and the Earth’s magnetic field), the orientations estimated by the KF are considered independent observations of the entries of $Q$. Put differently, the KF estimate $\hat{q}_i$ and its covariance matrix $P_i$ is taken as a rotation observation $z_i$ with covariance matrix $R_i$. It is modelled as

$$h_i(\hat{X}) = \hat{q}_i \oplus \delta \theta_i, \quad \delta \theta_i \sim \mathcal{N}(0, R_i).$$

The negative log-likelihood $\ell_i$ of a rotation observation is

$$\ell_i = \left( z_i \square h_i(\hat{X}) \right)^\top R_i^{-1} \left( z_i \square h_i(\hat{X}) \right),$$

where the predicted rotation observation $h(\hat{X}) = \hat{q}_i$ is the expected value of (20). Note that $\ell_i = 0$ until $\hat{X}$ is perturbed from its initial guess to accommodate the axis observations (Section III-C).

C. Axis Observations

The axes of planar surfaces are extracted from a LiDAR measurement by using a robust normal estimation algorithm [15]. The algorithm estimates an axis at each point by fitting a plane to a subset of its neighbouring points meeting minimum consistency and orthogonal distance criteria. Point axes are then clustered (with the $\square$ operator as the distance metric) by using a density-based clustering algorithm [16]. Note that the axes from points on physically different surfaces can be clustered together. The axial mean and covariance [17] of each cluster form the axis observations. In other words, each cluster generates an axis observation, which is a random axis $z_j$ with covariance matrix $R_j$. Each LiDAR measurement produces a small number of axis observations (one per surface orientation in the point cloud).

All axis observations from every LiDAR measurement are rotated to $F_g$ via the entries of $Q$. The resulting axes are clustered, whose axial means form $A$. The axis observations in the cluster making up an entry of $A$ are associated with that entry, and axis observations not in a cluster are considered outliers. In short, this step combines generating the map with data association. This approach is taken because of the surjective mapping from the planar surfaces in the environment to the entries of $A$; that is, any entry of $A$ can be measured at any orientation of the MP. As a result, one cannot simply search the neighbourhoo of an orientation for associations as in pose-graph SLAM. The major benefit of this approach is that loop closures frequently result from observations of different planar surfaces belonging to the same entry of $A$, regardless of their relative position.

Formally, an observation of the $j$-th axis in $A$ observed from the $i$-th orientation in $Q$ is an axis observation $z_j$ with covariance matrix $R_j$. It is modelled as

$$h_{i,j}(X) = C(q_i) n_j \oplus \delta \phi_j, \quad \delta \phi_j \sim \mathcal{N}(0, R_j).$$
where $n_j$ is the entry of $A$ with which $z_j$ is associated. The negative log-likelihood $\ell_{i,j}$ of an axis observation is

$$\ell_{i,j} = (z_j \triangleq h_{i,j} (\hat{X}))^\top R_j^{-1} (z_j \triangleq h_{i,j} (\hat{X})).$$

(23)

where the predicted observed axis $h_{i,j} (\hat{X}) = C (q_i) \hat{n}_j$ is the expected value of (22).

**D. Optimization**

Let $\ell_1, \ldots, \ell_p$ be the negative log-likelihoods of $p$ observations, where $\ell_i$ is calculated from (21) (a rotation observation) or (23) (an axis observation). Each $\ell_k$ has an observation $z_k$, an observation covariance matrix $R_k$, and a predicted observation $h_k(X)$. The optimal state estimate $X^*$ is determined by using least squares on a manifold, as described in [6]. In brief,

$$X^* = \arg \min_X \sum_{i=1, \ldots, p} \ell_k.$$

(24)

The Levenberg-Marquardt (LM) algorithm—modified to use manifold operations—is used to solve (24). It iteratively solves for the optimal state perturbation $\Delta X^*$ and applies it to the state (via $\oplus$) if it reduces the negative log-likelihood of the observations. Jacobians are determined numerically and heuristics introduced by Nielsen [18] are used to choose the damping parameter and stopping conditions. After each iteration, if the state has not converged, data association is redone by re-clustering the axis observations (Section III-C). Finally, the optimal OP $Q^*$ is then used to rotate all axis observations to $\mathcal{F}_y$, producing an AM of the environment.

**IV. EXPERIMENTAL RESULTS**

The 3DAM algorithm was tested by using outdoor data collected at a roadside rock face near Ivy Lea, ON, Canada. A handheld device rigidly connecting a Microstrain 3DM-GX3-25 IMU and Microsoft Kinect v2—which is a time-of-flight camera—was used to scan an area measuring approximately $5 \times 2.5$ m. 3DAM data collection is pictured in Fig. 6. Hand measurements of the axes of the planar surfaces of the same rock face were also taken using a Brunton compass (a combined compass and inclinometer). This type of manual measurement is typical of geologists and geotechnical engineers.

The results in this section are portrayed using stereonets, which is the standard tool used to visualize the orientation of discontinuities in rock masses. Stereonets can be viewed as polar plots, where the angle corresponds to the dip direction (angle relative to North), and the radius corresponds to the dip (angle relative to the horizontal). A unit axis $n = [\kappa_1 \kappa_2 \lambda]^\top$ is related to dip direction and dip by

$$\text{dip direction} = \text{atan2} (\kappa_2, \kappa_1) + \pi,$$

$$\text{dip} = \frac{\pi}{2} - \text{atan2} \left( \lambda, \sqrt{\kappa_1^2 + \kappa_2^2} \right).$$

(25)

The stereonets generated by both the hand measurements and by the 3DAM algorithm are illustrated in Fig. 7 and a summary of the results is shown in Table I. It is important to note that the hand measurements should not be taken as ground truth; rather, they represent the commonly used approach to generate a stereonet. Such measurements are subject to (often unintentional) operator bias and uncertainty that is not typically recorded or easily quantifiable.

Some of the advantages of 3DAM over hand measure-
ments are evident from the results of Table I, namely: data collection is much faster (and less physically demanding), coverage of the rock face (i.e., number of axis observations) is more extensive, and measurement uncertainty is quantified (which is not the case for manual measurements). The joint sets estimated by the two methods (i.e., the clusters of points in the stereonets in Fig. 7) are similar for the two methods, and (b) 3DAM. The contours depict the relative concentration of axes per unit area of the stereonet.

V. CONCLUSION

This paper provides a new mobile solution to the 3D axis mapping problem and describes an example application, with accompanying field results. Contributions include:

- A geometric description of unit axes and axis vectors, the axis product, and introduction of a common notation with rotations (Section II-B). These parameterizations follow in part from the work by Hertzberg et al. [6].
- The 3DAM algorithm and how it differs compared to traditional graph-based SLAM to account for the

**REFERENCES**


Fig. 7. Stereonets of a rock face generated using (a) hand measurements, and (b) 3DAM. The contours depict the relative concentration of axes per unit area of the stereonet.