

Bingham Distribution-Based Linear Filter for Online Pose Estimation

Rangaprasad Arun Srivatsan, Mengyun Xu, Nicolas Zavallos and Howie Choset

Robotics Institute, Carnegie Mellon University,

5000 Forbes Avenue, Pittsburgh, PA 15213.

Email: (arangapr,mengyunx,nzevallo}@andrew.cmu.edu, choset@cs.cmu.edu

Abstract—Pose estimation is central to several robotics applications such as registration, hand-eye calibration, SLAM, etc. Online pose estimation methods typically use Gaussian distributions to describe the uncertainty in the pose parameters. Such a description can be inadequate when using parameters such as unit-quaternions that are not unimodally distributed. A Bingham distribution can effectively model the uncertainty in unit-quaternions, as it has antipodal symmetry and is defined on a unit-hypersphere. A combination of Gaussian and Bingham distributions is used to develop a linear filter that accurately estimates the distribution of the pose parameters, in their true space. To the best of our knowledge our approach is the first implementation to use a Bingham distribution for 6 DoF pose estimation. Experiments assert that this approach is robust to initial estimation errors as well as sensor noise. Compared to state of the art methods, our approach takes fewer iterations to converge onto the correct pose estimate. The efficacy of the formulation is illustrated with a number of simulated examples on standard datasets as well as real-world experiments.

I. INTRODUCTION

Several applications in robotics require estimation of pose (translation and orientation) between a model frame and a sensor frame, for example, medical image registration [21], manipulation [5], hand-eye calibration [6] and navigation [5]. Filtering-based online pose estimation techniques have particularly been a popular choice due to their ability to adapt to noisy sensor measurements. Most of the current filtering methods use unimodal Gaussian distribution for modeling the uncertainty in the pose parameters. Such a distribution is a good choice to capture the uncertainty in parameters that are defined in a Euclidean space. However, the uncertainty in parameters such as unit-quaternions when modeled using a Gaussian does not consider the structure of the underlying space, i.e., antipodal symmetry introduced by $\tilde{q} = -\tilde{q}$ [18]. This work introduces an online pose estimation method that uses a combination of Bingham and Gaussian distribution to accurately and robustly estimate the pose.

Most of the prior work on online pose estimation linearize the non-linear measurement model. This results in inaccurate estimates especially when the initial pose estimate is erroneous. In order to address this issue, recently Srivatsan *et al.* [29] used dual-quaternions and developed a linear Kalman filter which is robust to initial pose errors. In this work, a Bingham distribution is used to model the uncertainty in the pose parameters, and a linear measurement model is adapted from [29] to develop a linear filter for online pose estimation. The Bingham distribution is defined on a unit hypersphere and captures the bimodality of the unit-quaternions [3] (see Fig. 1). When compared to prior methods, the use of the Bingham

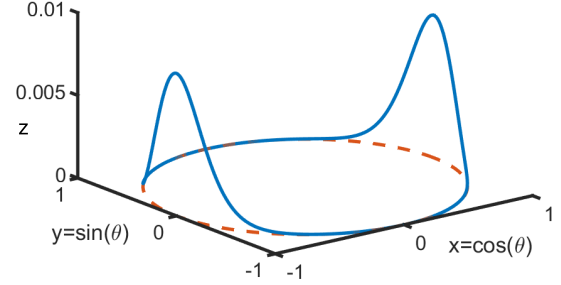


Fig. 1. A 2D Bingham distribution: $z = \frac{1}{N} \exp(\mathbf{s}^T \mathbf{M} \mathbf{Z} \mathbf{M}^T \mathbf{s})$, where $\mathbf{M} = \mathbf{I}_{2 \times 2}$, $\mathbf{Z} = \text{diag}(0, -10)$, and $\mathbf{s} = (x, y)$. The mode is at $x = \pm 1, y = 0$. More details can be obtained from Sec. III-B.

distribution results in a more principled formulation that has lower computation time, because there is no normalization step or projection onto a hyper-sphere. Another advantage of our approach compared to existing methods [29, 21, 22] is the ability to update the pose using surface-normal measurements as well as simultaneous multiple measurements (as obtained from a stereo camera or lidar).

Inspired by [14, 29], the pose is estimated by decoupling orientation from translation estimation, in this work. The method uses a Bingham distribution-based filtering (BF) for orientation estimation and a Gaussian distribution for translation estimation. While there has been some recent work on using the BF for orientation estimation [18, 9], there are some key differences compared to our approach. Firstly, prior work assumes that the state and measurements both are unit quaternions. This is not true in our case, since our measurements are point locations. Secondly, prior works deal with non-linear measurement models, hence resorting to unscented filtering. This results in computation of the normalization constant which is known to be expensive [10, 18]. Our approach uses a linear model (as developed by Srivatsan *et al.* [29]) and hence bypasses the computation of normalization constant.

In this work, we systematically estimate the pose for cases with position measurements as well as surface-normal measurements. In Sec. V comparison to state-of-the-art methods are provided. Our approach produces accurate estimates even in the presence of high initial errors and sensor noise, with fewer iterations. Finally, this work focuses on static pose estimation, but the ideas presented can be easily adapted to dynamic pose estimation. More details on this can be obtained from Sec. VI.

II. RELATED WORK

Batch Processing Approaches

Pose estimation has been of interest for a long time in the robotics literature. Much of the early literature deals with collecting all sensor measurements and processing them offline in a batch to estimate the pose. Horn *et al.* [14] developed a least squares implementation for pose estimation with known point correspondence. Besl *et al.* [1] introduced the iterative closest point (ICP), which extends Horn's methods for unknown point correspondence by iteratively estimating point correspondence and performing least squares optimization. Several variants to the ICP have been developed [25, 32]. Thrun *et al.* [26] further generalized the ICP by incorporating measurement noise uncertainties. Taylor *et al.* [2] have recently developed a probabilistic framework to estimate pose using surface-normal in addition to position measurements, while incorporating measurement uncertainty similar to [26].

Probabilistic Sequential Estimation

Probabilistic sequential estimation approaches provide sequential state updates based on a continuous stream of sensor measurements. The uncertainty in the state variables is often modeled using probability distributions functions (pdf) and the parameters of the pdf are updated after each measurement. In contrast to batch estimation methods, where there is no indication of when to stop collecting measurements, convergence of the state estimate and decrease in the state uncertainty provides clear indication of when to stop collecting measurements

Gaussian Filtering Approaches: Several sequential estimation methods are based on Kalman filters, which model the states and measurements using Gaussian distributions [22, 21, 13, 29]. Kalman filters by construction provide optimal state estimates when the process and measurement models are linear and the states and measurements are Gaussian distributed [15]. Pose estimation is inherently a non-linear problem, and hence linear Kalman filters produce poor estimates [13]. Several variants of the Kalman filter have been introduced to handle the non-linearity. EKF-based filters perform first-order linear approximations of the non-linear models and produce estimates which are known to diverge in the presence of high initial estimation errors [21]. UKF-based methods utilize evaluation at multiple points, which can be expensive for a high-dimensional system such as $SE(3)$. UKF-based methods also require tuning a number of parameters, which can be unintuitive.

Non-Gaussian Filtering Approaches: There has been some recent work in robotics towards the use of alternative distributions to model the noise on rotations for pose estimation problems. For example, Langevin distributions have been used for pose estimation [4, 24]. Henebeck *et al.* have recently developed a Bingham distribution-based recursive filtering approach for orientation estimation [9]. Glover *et al.* [10] use Bingham distribution to describe the orientation features, while Hanebeck *et al.* use this distribution for planar pose estimation [8]. Our work takes inspiration from these works

for modeling the uncertainty in the orientation using Bingham distribution. The use of Bingham distribution to model uncertainties in rotation parameters is a very valuable tool that has been largely under-utilized by the robotics community, as also noted by [10]. One of the important reasons for this, is the difficulty in computing the normalization constant as well as performing expensive convolution operation over the distributions [3]; which are both avoided in our approach. To the best of our knowledge our approach is the first of its kind that uses the BF for 6 DoF pose estimation.

Alternate Parameterizations for Filtering: Prior work also has looked at several parameterizations of $SE(3)$ that would improve the performance of the filters. In [13] the state variables are confined over a known Riemannian manifold and a UKF is used to estimate the pose. Quaternions are used to parametrize $SO(3)$ and the state is estimated using an EKF in [20] and UKF in [19]. An iterated EKF with dual quaternions to parameterize the pose has been used in [11].

Linear Filtering Approach: Srivatsan *et al.* [29] have recently developed a linear Kalman filter for pose estimation using dual quaternions and pairwise measurement update. While this method has been shown to be robust to errors in initial state estimate and sensor noise, it has a few drawbacks: (1) The uncertainty in the quaternions used for orientation estimate were modeled using Gaussians which do not consider the condition, $\tilde{\mathbf{q}}$ and $-\tilde{\mathbf{q}}$ represent the same rotation. (2) The filter by itself does not produce unit-quaternion estimates and hence after each estimate, a projection step is used to normalize the state. Such a projection would be difficult to implement if the estimated state had a near zero norm. (3) The approach only performs pairwise measurement update. However, in many practical applications such as image registration, several ($\approx 10^4$) measurements are available for processing in each update step; and a pairwise update could be very inefficient and time consuming.

III. MATHEMATICAL BACKGROUND

Before going into the description of the linear filter for pose estimation, we provide a brief introduction to the concepts of quaternions and the Bingham distribution.

A. Quaternions

While there are many representations for $SO(3)$ elements such as Euler angles, Rodrigues parameters, axis angles, etc, in this work we use unit-quaternions. We prefer the quaternions because their elements vary continuously over the unit sphere S^3 as the orientation changes, avoiding discontinuous jumps (inherent to three-dimensional parameterizations).

A quaternion $\tilde{\mathbf{q}}$ is a 4-tuple:

$$\tilde{\mathbf{q}} = (q_0, q_1, q_2, q_3), \quad \tilde{\mathbf{q}} \in \mathbb{R}^4,$$

where q_0 is the scalar part and $\mathbf{q} = (q_1, q_2, q_3)^T = \text{vec}(\tilde{\mathbf{q}})$ is the vector part of the quaternion. Sometimes an alternate convention is used where $\tilde{\mathbf{q}} = (q_1, q_2, q_3, q_0) = (\text{vec}(\tilde{\mathbf{q}}), \text{scalar}(\tilde{\mathbf{q}}))$ [3].

1) *Quaternion Multiplication*: Multiplication of two quaternions $\tilde{\mathbf{p}}$ and $\tilde{\mathbf{q}}$ is given by

$$\tilde{\mathbf{p}} \odot \tilde{\mathbf{q}} = \begin{bmatrix} p_0 & -\mathbf{p}^T \\ \mathbf{p} & \mathbf{p}^\times + p_0 \mathbf{I}_3 \end{bmatrix} \mathbf{q} = \begin{bmatrix} q_0 & -\mathbf{q}^T \\ \mathbf{q} & -\mathbf{q}^\times + q_0 \mathbf{I}_3 \end{bmatrix} \mathbf{p} \quad (1)$$

where \odot is the quaternion multiplication operator and $[\mathbf{v}]^\times$ is the skew-symmetric matrix formed from the vector \mathbf{v} .

2) *Quaternion Conjugate*: Given a quaternion $\tilde{\mathbf{q}}$, its conjugate $\tilde{\mathbf{q}}^*$ can be written as: $\tilde{\mathbf{q}}^* = (q_0, -q_1, -q_2, -q_3)$. The norm of a quaternion is $\|\tilde{\mathbf{q}}\| = \sqrt{\text{scalar}(\tilde{\mathbf{q}} \odot \tilde{\mathbf{q}}^*)}$.

3) *Unit Quaternions*: A unit quaternion is one with $\|\tilde{\mathbf{q}}\| = 1$. Unit quaternions can be used to represent rotation about an axis (denoted by the unit vector \mathbf{k}) by an angle $\theta \in [-\pi, \pi]$ as follows

$$\tilde{\mathbf{q}} = \left(\cos\left(\frac{\theta}{2}\right), \mathbf{k} \sin\left(\frac{\theta}{2}\right) \right). \quad (2)$$

Since rotating about \mathbf{k} axis by θ is the same as rotating about $-\mathbf{k}$ axis by $-\theta$, $\tilde{\mathbf{q}}$ and $-\tilde{\mathbf{q}}$ both represent the same rotation. A point \mathbf{b} can be rotated by a quaternion $\tilde{\mathbf{q}}$ to obtain a new point \mathbf{a} as shown,

$$\tilde{\mathbf{a}} = \tilde{\mathbf{q}} \odot \tilde{\mathbf{b}} \odot \tilde{\mathbf{q}}^*, \quad (3)$$

where $\tilde{\mathbf{a}} = (0, \mathbf{a})$ and $\tilde{\mathbf{b}} = (0, \mathbf{b})$ are quaternion representations of \mathbf{a}, \mathbf{b} respectively.

B. Bingham distribution

The Bingham distribution was introduced in [3] as an extension of the Gaussian distribution, conditioned to lie on the surface of a unit hyper-sphere. The Bingham distribution is widely used to analyze paleomagnetic data [17], computer vision [12] and directional statistics [3] and recently in robotics [18, 9, 10, 8].

Definition 1. Let $\mathcal{S}^{d-1} = \{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x}\| = 1\} \subset \mathbb{R}^d$ be the unit hypersphere in \mathbb{R}^d . The probability density function $f : \mathcal{S}^{d-1} \rightarrow \mathbb{R}$ of a Bingham distribution is given by

$$f(\mathbf{x}) = \frac{1}{N} \exp(\mathbf{x}^T \mathbf{M} \mathbf{Z} \mathbf{M}^T \mathbf{x}),$$

where $\mathbf{M} \in \mathbb{R}^{d \times d}$ is an orthogonal matrix ($\mathbf{M} \mathbf{M}^T = \mathbf{M}^T \mathbf{M} = \mathbf{I}^{d \times d}$), $\mathbf{Z} = \text{diag}(0, z_1, \dots, z_{d-1}) \in \mathbb{R}^{d \times d}$ with $0 \geq z_1 \geq \dots \geq z_{d-1}$ is known as the concentration matrix, and N is a normalization constant.

1) *Mode of the Distribution*: It can be shown that adding a multiple of the identity matrix $\mathbf{I}^{d \times d}$ to \mathbf{Z} does not change the distribution [3]. Thus, we conveniently force the first entry of \mathbf{Z} to be zero [3]. Because it is possible to swap columns of \mathbf{M} and the corresponding diagonal entries in \mathbf{Z} without changing the distribution, we can enforce $z_1 \geq \dots \geq z_{d-1}$. This representation allows us to obtain the mode of the distribution very easily by taking the first column of \mathbf{M} . Note that sometimes an alternate convention is used in literature, wherein \mathbf{Z} is chosen such that the last entry of \mathbf{Z} is 0 and the last column of \mathbf{M} is chosen as the mode of the distribution [18, 3].

2) *Normalization Constant*: The normalization constant N is given by

$$N = \int_{\mathcal{S}^{d-1}} \exp(\mathbf{x}^T \mathbf{Z} \mathbf{x}) d\mathbf{x},$$

The matrix \mathbf{M} is not a part of the normalization constant, because $N(\mathbf{Z}) = N(\mathbf{M} \mathbf{Z} \mathbf{M}^T)$ [3]. Computation of the normalization constant is difficult and often one resorts to some form of approximation such as saddle point approximations, or precomputed lookup tables (see [10] and the references therein).

3) *Antipodal Symmetry*: An example of the PDF for two dimensions ($d = 2$) is shown in Fig. 1. The PDF is antipodally symmetric, i.e., $f(\mathbf{x}) = f(-\mathbf{x})$ holds for all $\mathbf{x} \in \mathcal{S}^{d-1}$. The antipodal symmetry is important when dealing with distribution of unit-quaternions, because the $\tilde{\mathbf{q}}$ and $-\tilde{\mathbf{q}}$ describe the same rotation. The Bingham distribution with $d = 4$ is used to describe the uncertainty in the space of the unit-quaternions.

4) *Product of two Bingham Distributions*: Similar to a Gaussian, the product of two Bingham PDFs is a Bingham distribution, which can be rescaled to form a PDF [18]. Consider two Bingham distributions $f_i(\mathbf{x}) = \frac{1}{N_i} \exp(\mathbf{x}^T \mathbf{M}_i \mathbf{Z}_i \mathbf{M}_i^T \mathbf{x})$, $i = 1, 2$. Then,

$$\begin{aligned} f_1(\mathbf{x}) \cdot f_2(\mathbf{x}) &= \frac{1}{N_1 N_2} \exp(\mathbf{x}^T \underbrace{(\mathbf{M}_1 \mathbf{Z}_1 \mathbf{M}_1^T + \mathbf{M}_2 \mathbf{Z}_2 \mathbf{M}_2^T)}_{\mathbf{A}} \mathbf{x}) \\ &\propto \frac{1}{N} \exp(\mathbf{x}^T \mathbf{M} \mathbf{Z} \mathbf{M}^T \mathbf{x}), \end{aligned} \quad (4)$$

where N is the new normalization constant after renormalization, \mathbf{M} is composed of the unit eigenvectors of \mathbf{A} . $\mathbf{Z} = \mathbf{D} - \mathbf{D}_{11} \mathbf{I}_{d \times d}$ where \mathbf{D} has the eigenvalues of \mathbf{A} (sorted in descending order) and \mathbf{D}_{11} refers to the largest eigenvalue.

5) *Calculating the Covariance*: Even though a Bingham distributed random vector \mathbf{x} only takes values on the unit hyper-sphere, it is still possible to compute a covariance matrix in \mathbb{R}^d , which is given by: $\text{Cov}(\mathbf{x}) = E(\mathbf{x}^2) - E(\mathbf{x})^2$ [18]. Upon simplification one obtains

$$\text{Cov}(\mathbf{x}) = -0.5 \left(\mathbf{M}(\mathbf{Z} + c\mathbf{I})\mathbf{M}^T \right)^{-1}, \quad (5)$$

where $c \in \mathbb{R}$ can be arbitrarily chosen as long as $(\mathbf{Z} + c\mathbf{I})$ is negative definite [9]. Without loss of generality $c = \min(z_i)$ is chosen in this work.

6) *Composition of two Bingham Distributions*: Composition can be useful when we want to disturb a system whose uncertainties are modeled with a Bingham distribution with a Bingham distributed noise. Unfortunately, the Bingham distribution is not closed under composition and we can only approximate the composition as a Bingham [3]. Given two Bingham distributions f_1 and f_2 , their covariance matrices $\text{Cov}_1, \text{Cov}_2$ are first composed to obtain Cov_3 . The parameters of the approximate Bingham distribution obtained by composition, are then calculated using Eq. 5 (More details can be obtained from [18]).

IV. PROBLEM FORMULATION

In this work we consider pose estimation applications that use— 1) position measurements and 2) position and surface-normal measurements. The measurement model for both these cases are typically non-linear [2]. Inspired by [29], we derive linear models for both these cases.

A. Position Measurements

Let $\mathbf{a}_i, \mathbf{b}_i \in \mathbb{R}^3$, ($i = 1, \dots, n$) be the locations of n points in two different reference frames whose relative pose is to be estimated. The relation between points \mathbf{a}_i and \mathbf{b}_i , is given by

$$\mathbf{a}_i = \mathbf{R}\mathbf{b}_i + \mathbf{t}, \quad i = 1, \dots, n, \quad (6)$$

where $\mathbf{R} \in SO(3)$ and $\mathbf{t} \in \mathbb{R}^3$. In an application such as point-registration, \mathbf{a}_i are points in CAD-model frame and \mathbf{b}_i are points in sensor frame respectively.

1) *Update Model*: First consider the scenario where points in the sensor frame are obtained one at a time in a sequential manner, as typically observed in the case of robotic probing [28]. Similar to [29], the equations for updating the pose estimate given a pair of measurements ($n = 2$), are derived. From Eq. 3, Eq. 6 can be rewritten as

$$\tilde{\mathbf{a}}_1 = \tilde{\mathbf{q}} \odot \tilde{\mathbf{b}}_1 \odot \tilde{\mathbf{q}}^* + \tilde{\mathbf{t}}, \quad (7)$$

$$\tilde{\mathbf{a}}_2 = \tilde{\mathbf{q}} \odot \tilde{\mathbf{b}}_2 \odot \tilde{\mathbf{q}}^* + \tilde{\mathbf{t}}, \quad (8)$$

where $\tilde{\mathbf{q}}$ is as defined in Eq. 2 and $\tilde{\mathbf{t}} = (0, \mathbf{t})$. Subtracting Eq. 8 from Eq. 7,

$$\begin{aligned} \tilde{\mathbf{a}}_1 - \tilde{\mathbf{a}}_2 &= \tilde{\mathbf{q}} \odot (\tilde{\mathbf{b}}_1 - \tilde{\mathbf{b}}_2) \odot \tilde{\mathbf{q}}^*, \\ \Rightarrow (\tilde{\mathbf{a}}_1 - \tilde{\mathbf{a}}_2) \odot \tilde{\mathbf{q}} &= \tilde{\mathbf{q}} \odot (\tilde{\mathbf{b}}_1 - \tilde{\mathbf{b}}_2), \end{aligned} \quad (9)$$

since $\tilde{\mathbf{q}}$ is a unit-quaternion. Using matrix form of quaternion multiplication shown in Eq. 1, Eq. 9 can be rewritten as

$$\mathbf{H}\mathbf{q} = \mathbf{0}, \quad \text{where} \quad (10)$$

$$\mathbf{H} = \begin{bmatrix} 0 & -(\mathbf{a}_v - \mathbf{b}_v)^T \\ (\mathbf{a}_v - \mathbf{b}_v) & (\mathbf{a}_v + \mathbf{b}_v)^\times \end{bmatrix} \in \mathbb{R}^{4 \times 4}, \quad (11)$$

$\mathbf{a}_v = \mathbf{a}_1 - \mathbf{a}_2$ and $\mathbf{b}_v = \mathbf{b}_1 - \mathbf{b}_2$. Notice that Eq. 10 is a linear equation in terms of \mathbf{q} and is independent of \mathbf{t} . Upon obtaining the \mathbf{q} which lies in the null space of \mathbf{H} , \mathbf{t} can be obtained by adding Eq. 7 and Eq. 8,

$$\begin{aligned} \tilde{\mathbf{a}}_1 + \tilde{\mathbf{a}}_2 &= \tilde{\mathbf{q}} \odot (\tilde{\mathbf{b}}_1 + \tilde{\mathbf{b}}_2) \odot \tilde{\mathbf{q}}^* + 2\tilde{\mathbf{t}}, \\ \Rightarrow \mathbf{t} &= \frac{(\tilde{\mathbf{a}}_1 + \tilde{\mathbf{a}}_2) - \tilde{\mathbf{q}} \odot (\tilde{\mathbf{b}}_1 + \tilde{\mathbf{b}}_2) \odot \tilde{\mathbf{q}}^*}{2}. \end{aligned} \quad (12)$$

Eq. 10 and Eq. 12 were derived in [29] using dual quaternions, however, no geometrical intuition was provided. Fig. 2 provides the geometrical intuition behind the decoupled estimation of \mathbf{q} and \mathbf{t} . Estimating the pose between \mathbf{a}_i and \mathbf{b}_i , can be reduced to first estimating the orientation of vectors \mathbf{a}_v^{ij} and \mathbf{b}_v^{ij} and then estimating the translation between the centroids of the points. A similar idea is commonly used in Horn's method [14]. A key difference is that instead of forming vectors $\mathbf{a}_v = \mathbf{a}_1 - \mathbf{a}_2$ and $\mathbf{b}_v = \mathbf{b}_1 - \mathbf{b}_2$, Horn's method uses

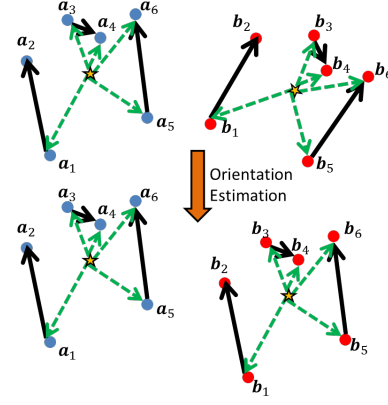


Fig. 2. Blue points (left) indicate \mathbf{a}_i and red points (right) indicate \mathbf{b}_i . Our approach constructs vectors $\mathbf{a}_v^{ij} = (\mathbf{a}_i - \mathbf{a}_j)$ and $\mathbf{b}_v^{ij} = (\mathbf{b}_i - \mathbf{b}_j)$ as shown by black arrows. The Bingham filter estimates the orientation between the black vectors. Horn's method [14] on the other hand, finds the orientation between the green dashed vectors. While the black vectors can be constructed online as the point measurements are received sequentially, the green-dashed vectors can be constructed only after all the data is collected.

$\mathbf{a}_v = \mathbf{a}_1 - \mathbf{a}_2$ and $\mathbf{b}_v = \mathbf{b}_1 - \mathbf{b}_2$, where \mathbf{a}_c and \mathbf{b}_c are the centroids of \mathbf{a}_i and \mathbf{b}_i respectively.

Further, Eq. 10 is similar to the one used by Hebert *et al.* [7]. However, in [7] uncertainties in sensor measurements were not considered while estimating \mathbf{q} . In this work, we model the uncertainty in the sensor measurements $\mathbf{a}_i, \mathbf{b}_i$ using Gaussian distribution. Let $\mathbf{a}_i^s = \mathbf{a}_i + \delta\mathbf{a}_i$ and $\mathbf{b}_i^s = \mathbf{b}_i + \delta\mathbf{b}_i$, where $(\cdot)^s$ is a sensor measurement, and $\delta(\cdot)$ is the noise as sampled from a zero mean Gaussian. Eq. 10 can be rewritten as

$$\begin{aligned} \mathbf{H}(\mathbf{a}_1, \mathbf{a}_2, \mathbf{b}_1, \mathbf{b}_2)\mathbf{q} &= \mathbf{0}, \\ \mathbf{H}(\mathbf{a}_1^s, \mathbf{a}_2^s, \mathbf{b}_1^s, \mathbf{b}_2^s)\mathbf{q} + \mathbf{G}(\mathbf{q})\boldsymbol{\mu} &= \mathbf{0}, \end{aligned} \quad (13)$$

where $\boldsymbol{\mu} = (\delta\mathbf{a}_1, \delta\mathbf{a}_2, \delta\mathbf{b}_1, \delta\mathbf{b}_2)^T$ and the expression for $\mathbf{G}(\mathbf{q})$ can be obtained from Eq. 37 of [29]. It can be shown that $\mathbf{G}(\mathbf{q})\boldsymbol{\mu}$ is a zero mean Gaussian noise, $\mathcal{N}(0, \mathbf{Q})$, where the uncertainty \mathbf{Q} is obtained analytically as shown in Sec. IV C of [29].

2) *Linear Filter*: In order to obtain an estimate of \mathbf{q} from Eq. 10, we use a Bingham distribution to model the uncertainty in \mathbf{q} ,

$$p(\mathbf{q}) = \frac{1}{N_1} \exp(\mathbf{q}^T \underbrace{\mathbf{M}_{k-1} \mathbf{Z}_{k-1} \mathbf{M}_{k-1}^T}_{\mathbf{D}_1} \mathbf{q}). \quad (14)$$

If the pose was changing with time, then a suitable process model can be employed as shown in [9]. In this work, we focus on static pose estimation and hence do not need a process model to evolve the pose estimate over time. Position measurements are obtained, which are in turn used to update the pose estimate. The pose is updated once for every pair of measurements received. The following is the probability of obtaining a sensor measurement \mathbf{z}_k , given the state \mathbf{q}_k ,

$$p(\mathbf{z}_k | \mathbf{q}_k) = \frac{1}{N_2} \exp\left(-\frac{1}{2}(\mathbf{z}_k - h(\mathbf{q}_k))^T \mathbf{Q}_k^{-1}(\mathbf{z}_k - h(\mathbf{q}_k))\right), \quad (15)$$

where $h(\mathbf{q}_k)$ is the expected sensor measurement and \mathbf{Q}_k is the measurement uncertainty. From Eq. 13, the measurement is set to $\mathbf{z}_k = \mathbf{0}$ and measurement model is set to $h(\mathbf{q}_k) = \mathbf{H}\mathbf{q}_k$. Since $\mathbf{z}_k = \mathbf{0}$ is not a true measurement, it is often referred to as pseudo-measurement in literature [23].

Thus Eq. 15 can be rewritten for our case as,

$$\begin{aligned} p(\mathbf{z}_k|\mathbf{q}_k) &= \frac{1}{N_2} \exp\left(-\frac{1}{2}(\mathbf{H}\mathbf{q}_k)^T \mathbf{Q}_k^{-1}(\mathbf{H}\mathbf{q}_k)\right), \\ &= \frac{1}{N_2} \exp(\mathbf{q}_k^T \mathbf{D}_2 \mathbf{q}_k), \end{aligned}$$

where $\mathbf{D}_2 = \frac{1}{2}(-\mathbf{H}^T \mathbf{Q}_k^{-1} \mathbf{H})$. Since \mathbf{Q}_k is a positive definite matrix (as required by a Gaussian), \mathbf{D}_2 is a negative definite matrix. As a result, we obtain an important result: $p(\mathbf{z}_k|\mathbf{q}_k)$ is an unnormalized Bingham distribution in \mathbf{q}_k .

Assuming the measurements are all independent of each other, the updated state given the current state estimate and measurement can be obtained by applying Bayes rule

$$\begin{aligned} p(\mathbf{q}_k|\mathbf{z}_k) &\propto p(\mathbf{q}_k)p(\mathbf{z}_k|\mathbf{q}_k) \\ &\propto \frac{1}{N_1} \exp(\mathbf{q}_k^T \mathbf{D}_1 \mathbf{q}_k) \frac{1}{N_2} \exp(\mathbf{q}_k^T \mathbf{D}_2 \mathbf{q}_k) \quad (16) \end{aligned}$$

$$\propto \exp(\mathbf{q}_k^T \mathbf{M}_k \mathbf{Z}_k \mathbf{M}_k^T \mathbf{q}_k). \quad (17)$$

And thus it can be seen that $p(\mathbf{q}_k|\mathbf{z}_k)$ is a Bingham distribution, where $\mathbf{M}_k \mathbf{Z}_k \mathbf{M}_k^T$ is obtained from the product of Bingham's as shown in Eq. 4. As mentioned in Sec. III-B4, the mode of the distribution \mathbf{q}_k , is the first column of \mathbf{M}_k .

After updating \mathbf{q}_k , \mathbf{t}_k is calculated from Eq. 12. The uncertainty in \mathbf{t}_k can be calculated as shown in Sec. IV-C of [29]. Hence, the state is updated once for every pair of measurements received, until a convergence condition is reached, or maximum number of updates is reached.

3) *Simultaneous Multi-measurement Update*: So far we have considered only the case where the state is updated once per pair of measurements. However, such an approach can be inefficient when applied to pose estimation from stereo cameras or KinectTM. In such applications, one typically obtains several position measurements at each time instant and processing the measurements in a pairwise manner can be time consuming. In order to address this situation, we can rewrite Eq. 10 as:

$$\mathbf{H}_j \mathbf{q} = \mathbf{0}, \quad j = 1, \dots, m.$$

\mathbf{H}_j has the form as shown in Eq. 11, where $\mathbf{a}_v, \mathbf{b}_v$ are obtained from points-pairs constructed by subtracting random pairs of points or subtracting each point from the centroid (similar to [14]). Since the measurements are assumed to be independent, we have

$$\begin{aligned} p(\mathbf{z}_k|\mathbf{q}_k) &= \prod_{j=1}^m \frac{1}{N_2^j} \exp\left(\frac{-1}{2}(\mathbf{H}_j \mathbf{q}_k)^T \mathbf{Q}^{-1}(\mathbf{H}_j \mathbf{q}_k)\right), \\ &= \frac{1}{N_3} \exp(\mathbf{q}_k^T \mathbf{D}_3 \mathbf{q}_k), \end{aligned} \quad (18)$$

where $\mathbf{D}_3 = \frac{1}{2} \sum_j (-\mathbf{H}_j^T \mathbf{Q}^{-1} \mathbf{H}_j)$ and $N_3 = \prod_{j=1}^m N_2^j$. Eq. 16 can be rewritten as

$$\begin{aligned} p(\mathbf{q}_k|\mathbf{z}_k) &\propto \frac{1}{N_1} \exp(\mathbf{q}_k^T \mathbf{D}_1 \mathbf{q}_k) \frac{1}{N_3} \exp(\mathbf{q}_k^T \mathbf{D}_3 \mathbf{q}_k) \\ &\propto \exp(\mathbf{q}_k^T \mathbf{M}_k \mathbf{Z}_k \mathbf{M}_k^T \mathbf{q}_k), \end{aligned} \quad (19)$$

where $\mathbf{M}_k \mathbf{Z}_k \mathbf{M}_k^T$ is obtained from Bingham multiplication. \mathbf{q}_k and \mathbf{t}_k are obtained as shown in Sec. IV-A2.

B. Surface-normal Measurements

In some applications, in addition to position measurements, surface-normal measurements may also be available [28]. The following equation relates the surface-normals in the two frames,

$$\begin{aligned} \mathbf{n}_i^a &= \mathbf{R} \mathbf{n}_i^b, \quad i = 1, \dots, l \\ \Rightarrow \tilde{\mathbf{n}}_i^a &= \tilde{\mathbf{q}} \odot \tilde{\mathbf{n}}_i^b \odot \tilde{\mathbf{q}}^* \\ \Rightarrow \tilde{\mathbf{n}}_i^a \odot \tilde{\mathbf{q}} &= \tilde{\mathbf{q}} \odot \tilde{\mathbf{n}}_i^b \\ \Rightarrow \mathbf{G}_i \mathbf{q} &= \mathbf{0}, \quad \text{where} \\ \mathbf{G}_i &= \begin{bmatrix} 0 & -(\mathbf{n}_i^a - \mathbf{n}_i^b)^T \\ (\mathbf{n}_i^a - \mathbf{n}_i^b) & (\mathbf{n}_i^a + \mathbf{n}_i^b)^\times \end{bmatrix}, \end{aligned}$$

where \mathbf{n}_i^a are surface-normals in CAD-model frame and \mathbf{n}_i^b are surface-normals in the sensor frame. Similar to the derivation in the case of position measurements (see Eq. 18), we obtain,

$$p(\mathbf{z}_k|\mathbf{q}_k) = \frac{1}{N_4} \exp(\mathbf{q}_k^T \mathbf{D}_4 \mathbf{q}_k), \quad (20)$$

where $\mathbf{D}_4 = \frac{1}{2} \sum_i (-\mathbf{G}_i^T \mathbf{S}_k^{-1} \mathbf{G}_i) + \frac{1}{2} \sum_j (-\mathbf{H}_j^T \mathbf{Q}^{-1} \mathbf{H}_j)$, \mathbf{S}_k is the measurement uncertainty of the pseudo-measurement. Thus, we have

$$\begin{aligned} p(\mathbf{q}_k|\mathbf{z}_k) &\propto \frac{1}{N_1} \exp(\mathbf{q}_k^T \mathbf{D}_1 \mathbf{q}_k) \frac{1}{N_4} \exp(\mathbf{q}_k^T \mathbf{D}_4 \mathbf{q}_k), \\ &\propto \exp(\mathbf{q}_k^T \mathbf{M}_k \mathbf{Z}_k \mathbf{M}_k^T \mathbf{q}_k). \end{aligned}$$

V. RESULTS

In this section, we consider two scenarios for using the Bingham distribution-based linear filter– 1) known correspondence and 2) unknown correspondence. Without loss of generality, we assume that there is no prior information about the pose and for all the experiments we choose the following values, $\mathbf{M}_0 = \mathbf{I}_{4 \times 4}$, $\mathbf{Z}_0 = \text{diag}(0, 1, 1, 1) \times 10^{-300}$ ¹, which represents an uninformative prior with high initial uncertainty. The measurements are modeled according to a Gaussian distribution with 0 mean and standard deviation of 0.2 mm. We restrict the maximum number of state updates to 100.

¹ 10^{-300} is the smallest positive normalized floating-point number in IEEE[®] double precision.

A. Known Correspondence

In this section, we assume that the correspondence between points $\mathbf{a}_i \in \mathbb{R}^3$ and $\mathbf{b}_i \in \mathbb{R}^3$ are known, and estimate the pose between the frames that these two point sets lie in. The coordinates of the data set \mathbf{a}_i is produced by drawing points uniformly in the interval $[-250 \text{ mm}, 250 \text{ mm}]$. To create the noiseless data set \mathbf{b}_i , a random transformation is applied to \mathbf{a}_i . This transformation is generated by uniformly drawing the rotational and translational parameters in the intervals $[-90^\circ, 90^\circ]$ and $[-90 \text{ mm}, 90 \text{ mm}]$, respectively. In Experiment 1, no noise is added to \mathbf{b}_i . In Experiment 2 and Experiment 3, a noise uniformly drawn from $[-2 \text{ mm}, 2 \text{ mm}]$ and $[-10 \text{ mm}, 10 \text{ mm}]$ respectively, is added to each coordinate of \mathbf{b}_i . Next, the linear Bingham filter (BF) is used to estimate the pose.

TABLE I

	RMS (mm) (Expt. 1)	RMS (mm) (Expt. 2)	RMS (mm) (Expt. 3)
BF	0.00	2.29	12.12
DQF	6.14	7.70	70.99
UKF	2.67	4.91	12.25
EKF	94.65	21.97	56.52

This procedure is repeated 500 times with different datasets and different transformation that are randomly generated. The results are compared with dual quaternion filter (DQF) [29], UKF [21] and EKF [22]. Table I shows the RMS errors for all the filtering methods considered ².

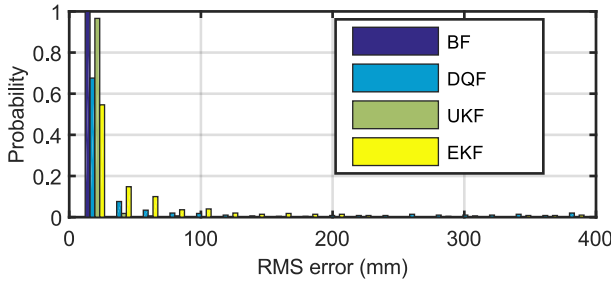


Fig. 3. Histogram shows the RMS errors for the Bingham filter (BF), dual quaternion filter (DQF), unscented Kalman filter (UKF) and extended Kalman filter (EKF). The results shown are for Expt. 3, where the sensed points have a noise uniformly drawn from $[-10 \text{ mm}, 10 \text{ mm}]$. The BF is most accurate with an average RMS error of 12.12 mm.

Fig. 3 shows the histogram of errors for Expt. 3. For Expt. 3, the average run time for the BF is 0.05s, compared to 0.04s of DQF, 2.23s of UKF and 0.10s of EKF. The BF always estimates the pose with the lowest RMS error. The RMS error of UKF is small, but larger than the BF and it takes much longer to estimate (≈ 50 times slower). DQF and EKF often get trapped in local minima, which has also been noted earlier in [21, 27]. The performance of EKF and DQF can be improved

²All the calculation are carried out using MATLAB R2015a software from MathWorks, running on a ThinkPad T450s computer with 8 GB RAM and intel i7 processor.

by adopting an approach similar to [27], but the estimation time would also increase.

The BF provides accurate estimates because the filter is defined in the true space of the pose parameters. DQF is also a linear filter, but it performances poorly when the state uncertainty is high. The BF is also faster than other EKF and UKF as it is a linear filter with no Jacobian or sigma point computations.

1) *Real-world Example: Registering Camera and Robot Frame:* Fig. 4 shows an arm of a da Vinci[®] surgical robot (Intuitive Surgical Inc., Mountain View, CA) mounted on a table, and a stereo camera (ELP-1MP2CAM001 Dual Lens) mounted on a rigid stand. The relative pose between the robot's frame and the camera's frame is fixed, and needs to be estimated. To estimate this pose, the robot is telemanipulated in arbitrary paths and the location of tip of the robot \mathbf{a}_i is computed in the camera frame by segmenting the tip from the stereo image and estimating its center. The position of the tip in the robot frame, \mathbf{b}_i is obtained from the kinematics of the robot. The pose between the points \mathbf{a}_i and \mathbf{b}_i can be obtained as shown in Sec. V-A.

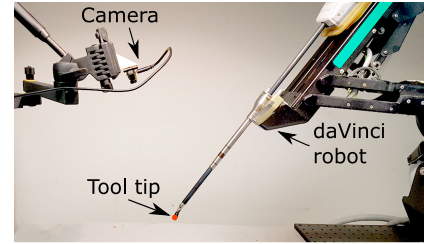


Fig. 4. A spherical tool tip is attached to the da Vinci robot. The tip is tracked using a stereo camera, which is held in a fixed position. As the robot is telemanipulated, the spherical tool-tip is tracked using the stereo camera, and the relative pose between the camera frame and the robot frame is estimated.

Table II shows the pose as estimated by our Bingham filtering approach using pairwise updates, using 20 simultaneous measurements per update (abbreviated as BFM in the Table) as well as by Horn's method. Fig. 5 shows the RMS error

TABLE II

	x (mm)	y (mm)	z (mm)	θ_x (rad)	θ_y (rad)	θ_z (rad)	Time (ms)	RMS (mm)
BF	28.91	-128.91	250.45	171.49	15.05	-144.76	25.8	8.93
BFM	6.27	-143.76	269.32	174.40	5.52	-139.25	2.1	5.91
Horn	2.66	-136.25	265.25	173.59	4.81	-141.75	56.8	4.88

versus number of measurements used. The BFM takes 40 measurements to converge as opposed to the BF and Horn's method, which take 90 measurements. The concentration matrix of the Bingham filter, \mathbf{Z} , provides the uncertainty in the state estimate, which serves as an indication for convergence of the state. Upon convergence, no more measurements need to be collected. There exists no such mechanism to indicate convergence in Horn's method. Hence Horn's method needs to be run repeatedly with all the measurements collected thus far.

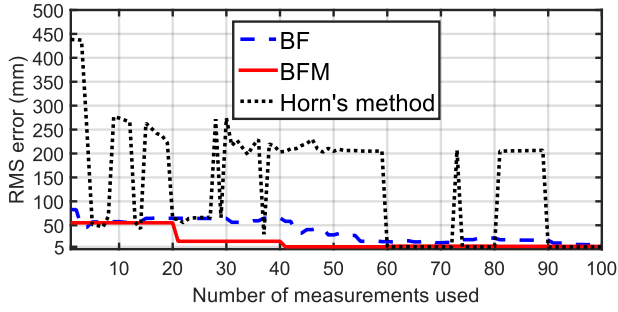


Fig. 5. Bingham filter using 20 simultaneous measurements per update (BFM) converges at 40 measurements. In comparison, Horn's method and Bingham filter with pairwise update (BF), both take ≈ 90 measurements to converge.

As a result the total run time till convergence of the BF and BFM is much lower than Horn's method. The RMS error of the BF is higher than the BFM, because multiple simultaneous measurements, help smooth out the effect of the noise in the position measurements.

B. Unknown Correspondence

In this section we assume that the points a_i and surface-normals n_i^a are the vertices and normals respectively, of a triangulated mesh. Fig. 6 shows the triangulated mesh in the shape of a bunny [31], which has 86,632 triangles.

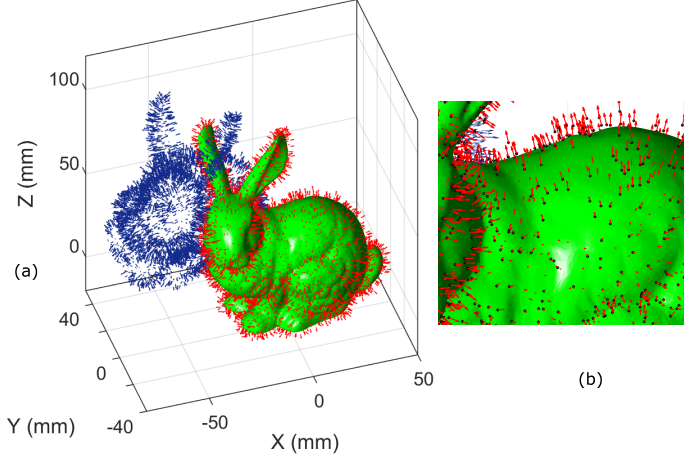


Fig. 6. (a) Triangulated mesh of Stanford bunny [31] is shown in green. Blue arrows represent initial location and red arrows represent estimated location of points and surface-normals. (b) Zoomed up view shows that the estimated location of points accurately rests on the triangulated mesh and the estimated direction of the surface-normals aligns well with the local surface normal. The Bingham filter takes 1.4s in MATLAB and 0.08s in C++ to estimate the pose.

We randomly pick 5000 points from the triangulated mesh and to each coordinate of the points, add a noise uniformly drawn from $[-2 \text{ mm}, 2 \text{ mm}]$. For each (b_i, n_i^b) , the correspondence is obtained by finding the closest point-normal pair (a_i, n_i^a) on the triangulated mesh. We estimate the pose using the BF with simultaneous multi-measurements as described

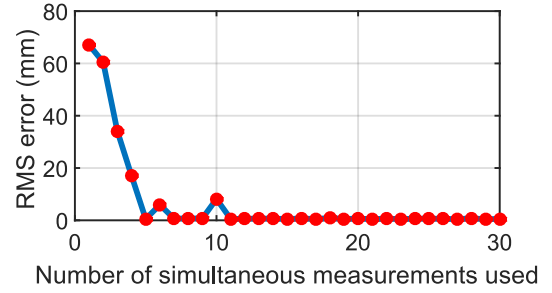


Fig. 7. Plot shows RMS error upon convergence versus number of simultaneous measurements used. The more the number of simultaneous measurements used, the lower is the RMS error.

in Sec. IV-A3. Fig. 7 shows the RMS error vs number of simultaneous measurements used. Update based on one pair of measurements results in a local optimum (RMS error is $\approx 70 \text{ mm}$ as shown in Fig. 7). However, the performance drastically improves when > 10 simultaneous measurements are used.

TABLE III

	x (mm)	y (mm)	z (mm)	θ_x (deg)	θ_y (deg)	θ_z (deg)	Time (s)	RMS (mm)
Actual	44.83	-50.45	7.15	-12.01	-21.49	-28.14	—	—
DQF	42.02	-53.95	6.63	-13.18	-19.86	-30.69	13.02	2.02
ICP	44.52	-49.16	6.32	-9.05	-19.11	-30.40	77.83	2.04
BFM	44.45	-50.38	7.65	-12.14	-21.75	-28.10	1.06	0.54
BFN	44.23	-50.44	7.21	-12.08	-21.23	-28.21	1.43	0.53

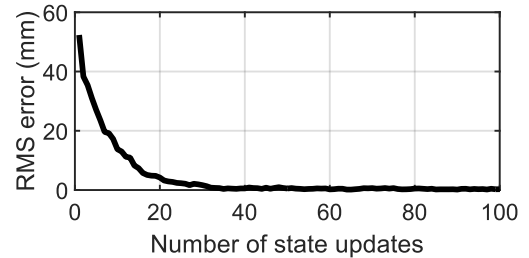


Fig. 8. Plot shows the RMS error in the pose vs number of state updates as estimated by the Bingham filter using 20 simultaneous position and normal measurements in each update. The estimate converges around 40 iterations.

The penultimate row of Table III shows the pose parameters as estimated by the BF using 20 simultaneous position measurements (This experiment is abbreviated as BFM). We also estimate the pose using 20 simultaneous surface-normal and position measurements (abbreviated as BFN in Table III). The RMS error for the BFN is slightly better than the BFM. However, the time taken by the BFN is slightly higher because surface-normals are used in addition to point locations when finding the correspondence. Fig. 6(a) shows the initial position of the surface-normals and point locations with blue arrows and the BFN estimated surface-normals and point locations with red arrow. The zoomed up image Fig. 6(b) shows that our approach accurately registers the points as well as aligns

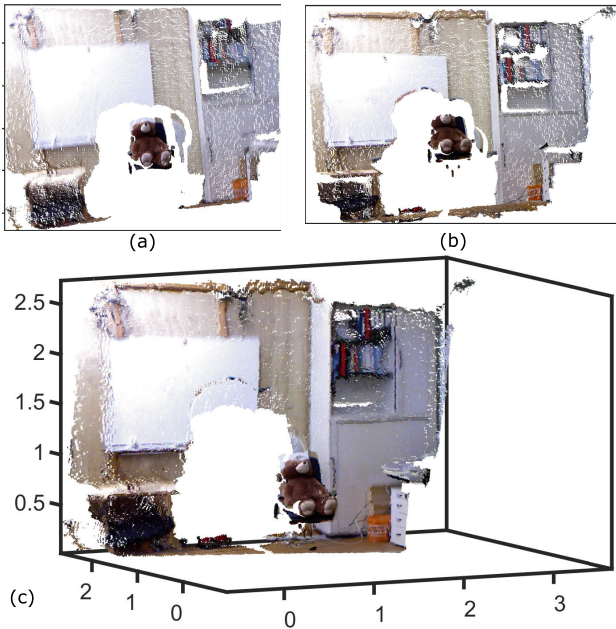


Fig. 9. (a), (b), are two RGB-D images obtained from KinectTM, with some overlapping region. (c) The point cloud model estimated by aligning the point clouds in (a) and (b) using the Bingham filter. The Bingham filter takes 0.21s to estimate the pose with an RMS error of 4.4cm, as opposed to ICP, which takes 0.46s with an RMS error of 6cm.

the surface-normals to the triangulated mesh. Table III also shows the pose parameters as estimated by ICP [1] and a modified version of DQF [27]. Our approach is orders of magnitude faster and more accurate than both these methods. Fig. 8 shows the RMS error at the end of each update step for BFN. The RMS error reduces to $< 0.6\text{mm}$ at around 40 state updates. To obtain the same accuracy as DQF and ICP ($\approx 2\text{mm}$), both BFM and BFN take ≈ 30 state updates, which takes 0.28s.

1) *Real-world Example– Point-cloud Stitching:* Stereo imaging devices such as the Microsoft KinectTM offer colored point cloud data (RGB-D: color and depth data), which is generated using a structured light based depth sensor. The KinectTM is widely used in robot navigation [16] and object manipulation [5]. In this work, we align a pair of point cloud data obtained from the KinectTM, using the Bingham filter, to develop a point-cloud model of the environment. It is assumed that there is some overlap between the two point clouds. We demonstrate our approach on RGB-D images taken from the ‘Freiburg1-Teddy’ dataset of [30]. Fig. 9(a), (b) shows the snapshots of the images. Fig. 9(d) shows the final model of the room as generated by our approach. We use 20 simultaneous measurements and the same initial conditions as in the previous cases. Our approach takes $\approx 0.21\text{s}$ for estimating pose, which is twice as fast as ICP which takes $\approx 0.46\text{s}$. In order to improve the speed we have implemented a C++ version of the Bingham filter, which takes only ≈ 2

ms³. The RMS error of our approach is 4.4cm, which is of the order of the accuracy of the sensor itself [16] and is better than the RMS error of ICP, 6cm.

VI. CONCLUSION AND DISCUSSIONS

In this work, a Bingham distribution-based linear filter (BF) was developed for online pose estimation. Bingham distribution captures the bimodal nature as well as unit norm constraint of the rotation quaternion. By adapting the linear measurement model developed by Srivatsan *et. al.* [29], a linear Bingham filter has been developed that updates the pose based on a pair of position measurements. Further the filter is extended to process surface-normal as well as multiple simultaneous measurements, for applications such as image registration and point-cloud stitching.

It has been shown through simulations and experiments that the BF is capable accurate pose estimation with less computation time compared to state-of-the-art methods. It is empirically observed that using multiple simultaneous measurements per update helps avoid local optima, when the correspondences are unknown. We also observe that position measurements reduce the RMS error to such an extent that using surface-normal measurements in addition offers very little improvement.

One drawback of our approach, as with most filtering based approaches, is that the estimate can be trapped in a local minima. This problem is more prevalent when point correspondences are unknown. Using a high initial uncertainty and more number of simultaneous measurements helps alleviate this problem. However, in some applications only pairs of measurements may be available per update, and the correspondences may be unknown (ex. robotic probing [28]). In such situations, better correspondences using a probabilistic metric as described in [2], can improve the estimate. Another approach to resolve this issue is to use a global optimizer for filtering-based methods such as [27].

In the future we plan to use the estimate of the concentration matrix of the Bingham distribution to guide where to collect the next set of measurements to improve the registration. While we limit ourselves to static pose estimation in this work, the approach can be easily adapted for dynamic pose estimation. If the sensor provides multiple position measurements at a high frequency rate, then a series of static online pose estimation can be performed to track the pose. Depending on the application, one could also develop a process model to capture the dynamics, and utilize an unscented Bingham filter [9] if this model is nonlinear.

ACKNOWLEDGMENTS

This work has been funded through the National Robotics Initiative by NSF grant IIS-1426655.

³Source code– https://github.com/biorobotics/bingham_registration/tree/ros-free

REFERENCES

- [1] Paul J Besl and Neil D McKay. Method for registration of 3-D shapes. In *Robotics-DL tentative*, pages 586–606. International Society for Optics and Photonics, 1992.
- [2] Seth D Billings, Emad M Bector, and Russell H Taylor. Iterative most-likely point registration (IMLP): A robust algorithm for computing optimal shape alignment. *PloS one*, 10(3):e0117688, 2015.
- [3] Christopher Bingham. An antipodally symmetric distribution on the sphere. *The Annals of Statistics*, pages 1201–1225, 1974.
- [4] Luca Carlone and Andrea Censi. From angular manifolds to the integer lattice: Guaranteed orientation estimation with application to pose graph optimization. *IEEE Transactions on Robotics*, 30(2):475–492, 2014.
- [5] Nikolas Engelhard, Felix Endres, Jürgen Hess, Jürgen Sturm, and Wolfram Burgard. Real-time 3D visual SLAM with a hand-held RGB-D camera. In *Proc. of the RGB-D Workshop on 3D Perception in Robotics at the European Robotics Forum, Vasteras, Sweden*, volume 180, 2011.
- [6] F. Faion, P. Ruoff, A. Zea, and U.D. Hanebeck. Recursive Bayesian calibration of depth sensors with non-overlapping views. In *15th International Conference on Information Fusion*, pages 757–762, July 2012.
- [7] Olivier D Faugeras and Martial Hebert. The representation, recognition, and locating of 3-d objects. *The international journal of robotics research*, 5(3):27–52, 1986.
- [8] Igor Gilitschenski, Gerhard Kurz, Simon J Julier, and Uwe D Hanebeck. A new probability distribution for simultaneous representation of uncertain position and orientation. In *17th International Conference on Information Fusion*, pages 1–7. IEEE, 2014.
- [9] Igor Gilitschenski, Gerhard Kurz, Simon J Julier, and Uwe D Hanebeck. Unscented orientation estimation based on the Bingham distribution. *IEEE Transactions on Automatic Control*, 61(1):172–177, 2016.
- [10] Jared Glover, Gary Bradski, and Radu Bogdan Rusu. Monte carlo pose estimation with quaternion kernels and the distribution. In *Robotics: Science and Systems*, volume 7, page 97, 2012.
- [11] James Samuel Goddard and Mongi A Abidi. Pose and motion estimation using dual quaternion-based extended Kalman filtering. In *Photonics West’98 Electronic Imaging*, pages 189–200. International Society for Optics and Photonics, 1998.
- [12] Tom SF Haines and Richard C Wilson. Belief propagation with directional statistics for solving the shape-from-shading problem. In *European Conference on Computer Vision*, pages 780–791. Springer, 2008.
- [13] Søren Hauberg, François Lauze, and Kim Steenstrup Pedersen. Unscented Kalman filtering on Riemannian manifolds. *Journal of Mathematical Imaging and Vision*, 46(1):103–120, 2013.
- [14] Berthold KP Horn. Closed-form solution of absolute orientation using unit quaternions. *JOSA A*, 4(4):629–642, 1987.
- [15] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering*, 82(1):35–45, 1960.
- [16] Kourosh Khoshelham and Sander Oude Elberink. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, 12(2):1437–1454, 2012.
- [17] Karsten Kunze and Helmut Schaeben. The Bingham distribution of quaternions and its spherical radon transform in texture analysis. *Mathematical Geology*, 36(8):917–943, 2004.
- [18] Gerhard Kurz, Igor Gilitschenski, Simon Julier, and Uwe D Hanebeck. Recursive estimation of orientation based on the Bingham distribution. In *16th International Conference on Information Fusion*, pages 1487–1494, 2013.
- [19] Joseph J LaViola Jr. A comparison of unscented and extended Kalman filtering for estimating quaternion motion. In *American Control Conference, 2003. Proceedings of the 2003*, volume 3, pages 2435–2440. IEEE, 2003.
- [20] João Luís Marins, Xiaoping Yun, Eric R Bachmann, Robert B McGhee, and Michael J Zyda. An extended Kalman filter for quaternion-based orientation estimation using MARG sensors. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 4, pages 2003–2011. IEEE, 2001.
- [21] Mehdi Hedjazi Moghari and Purang Abolmaesumi. Point-based rigid-body registration using an unscented Kalman filter. *IEEE Transactions on Medical Imaging*, 26(12):1708–1728, 2007.
- [22] Xavier Pennec and Jean-Philippe Thirion. A framework for uncertainty and validation of 3-D registration methods based on points and frames. *International Journal of Computer Vision*, 25(3):203–229, 1997.
- [23] PW Richards. Constrained Kalman filtering using pseudo-measurements. In *IEEE Colloquium on Algorithms for Target Tracking*, pages 75–79, 1995.
- [24] David M Rosen, Luca Carlone, Afonso S Bandeira, and John J Leonard. Se-sync: A certifiably correct algorithm for synchronization over the special euclidean group. *arXiv preprint arXiv:1612.07386*, 2016.
- [25] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the ICP algorithm. In *Proceedings of 3rd International Conference on 3-D Digital Imaging and Modeling*.
- [26] Aleksandr Segal, Dirk Haehnel, and Sebastian Thrun. Generalized-ICP. In *Robotics: Science and Systems*, volume 2, 2009.
- [27] Rangaprasad Arun Srivatsan and Howie Choset. Multiple start branch and prune filtering algorithm for nonconvex optimization. In *The 12th International Workshop on The Algorithmic Foundations of Robotics*. Springer, 2016.
- [28] Rangaprasad Arun Srivatsan, Elif Ayvali, Long Wang, Rajarshi Roy, Nabil Simaan, and Howie Choset. Complementary Model Update: A Method for Simultaneous

- Registration and Stiffness Mapping in Flexible Environments. In *IEEE International Conference on Robotics and Automation*, pages 924–930, 2016.
- [29] Rangaprasad Arun Srivatsan, Gillian T Rosen, Feroze D Naina, and Howie Choset. Estimating SE(3) elements using a dual quaternion based linear Kalman filter. In *Robotics : Science and Systems*, 2016.
- [30] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A Benchmark for the Evaluation of RGB-D SLAM Systems. In *Proc. of the International Conference on Intelligent Robot Systems*, Oct. 2012.
- [31] Greg Turk and Marc Levoy. The Stanford 3D Scanning Repository. *Stanford University Computer Graphics Laboratory* <http://graphics.stanford.edu/data/3Dscanrep>.
- [32] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *European Conference on Computer Vision*, pages 766–782. Springer, 2016.