Stereo Visual SLAM Based on Unscented Dual Quaternion Filtering

Simon Bultmann*, **Kailai Li** and **Uwe D. Hanebeck**
Intelligent Sensor-Actuator-Systems Laboratory (ISAS)
Institute for Anthropomatics and Robotics
Karlsruhe Institute of Technology (KIT), Germany
bultmann@ais.uni-bonn.de, kailai.li@kit.edu, uwe.hanebeck@ieee.org

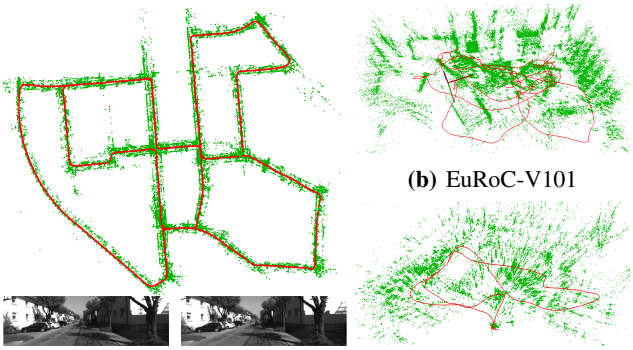
***Abstract*—We present DQV-SLAM (Dual Quaternion Visual SLAM). This novel feature-based stereo visual SLAM framework uses a stochastic filter based on the unscented transform and a progressive Bayes update, avoiding linearization of the nonlinear spatial transformation group. 6-DoF poses are represented by dual quaternions where rotational and translational components are stochastically modeled by Bingham and Gaussian distributions. Maps represented by point clouds of ORB-features are incrementally built and landmarks are updated with an unscented transform-based method. In order to get reliable measurements during the update, an optical flow-based approach is proposed to remove false feature associations. Drift is corrected by pose graph optimization once loop closure is detected. The KITTI and EuRoC datasets for stereo setup are used for evaluation. The performance of the proposed system is comparable to state-of-the-art optimization-based SLAM systems and better than existing filtering-based approaches.**

I. Introduction

Simultaneous Localization and Mapping (SLAM) entails parallel ego-motion tracking and building a map of unknown surroundings. Among different possible sensors, stereo cameras provide cost-efficient and informative perception. However, camera-only visual SLAM can be challenging, e.g., due to changing lighting conditions, blurring under fast motion, and dynamic scenarios with moving objects, which can lead to reduced accuracy.

The state-of-the-art visual SLAM frameworks typically rely on optimization, either for feature-based [1], [2] or direct methods [3], [4]. Optimization-based approaches are shown in [5] to provide better accuracy than filtering-based methods for the same computational cost, as they can process a large set of state variables more efficiently. However, stochastic filters are able to provide a probability distribution over the states without additional computational effort, which enables efficient obstacle avoidance and motion planning. For example, the safety margins for navigation can be adapted according to the current state uncertainty [6]. Moreover, SLAM systems employing a stochastic filter show comparably robust and accurate results with sparser extracted features [7]. This makes them a functional option besides the optimization-based approaches, in particular for mobile robots in practical applications.

*Simon Bultmann currently is with the Autonomous Intelligent Systems Group, Computer Science Institute VI, University of Bonn, Germany. This work was done during his time at ISAS, KIT.



(a) KITTI-00 **(b)** EuRoC-V101 **(c)** EuRoC-MH05

Fig. 1: Qualitative mapping (green point cloud) and tracking (red curve) results from the proposed SLAM framework with stereo setup onboard a vehicle (a) and drone (b, c).

Early filtering-based approaches to visual SLAM either use a particle filter [8] or the Extended Kalman Filter (EKF) [9] for pose and map estimation and are only able to operate in small-scale scenarios. Recently, filtering-based odometry or SLAM systems, which are capable of large-scale operation, were developed for visual-inertial sensor setups using an inertial measurement unit (IMU) in addition to the cameras [6], [7], [10]. ROVIO [7] employs an Iterated EKF and uses direct photometric image patch alignment while [10] develop a feature-based method employing a Multi-State Constraint Kalman Filter (MSCKF) [11]. In contrast to these systems, our approach only uses visual sensors, not requiring higher-order motion information, and employs a novel unscented transform-based nonlinear filter which avoids linearization.

Conventional filters for pose estimation rely on the local linearization of nonlinear spatial transformations as shown in [7], [12], where the EKF and Unscented Kalman Filter (UKF) are applied to the special Euclidean group SE(3) or SE(2). This is essentially an approximation under the assumption of small motion and low noise level. Besides, the SE(3) states are usually parameterized by the 4-dimensional transformation matrix, which suffers from overparameterization and numerical instability. Some orientation filters also use Euler angles, which can lead to ambiguity and gimbal lock. Dual quaternions [13], however, provide a natural parameterization of 6-DoF poses without ambiguity and with little redundancy.

In [14], an unscented orientation filter was first proposed based on Bingham-distributed unit quaternions. In [15], an unscented pose estimator was introduced for planar motion represented by dual quaternions. In [16], the first dual quaternion filter for 6-DoF pose estimation was proposed, where Bingham and Gaussian distributions are employed to model orientational quaternions and pure translation, respectively. Good efficiency as well as accuracy have been shown with experiments, and no local linearization is required. However, this filter imposes a linear formulation and a special metric based on point pairs, which restrict its application to static point cloud registration or stitching. The aforementioned works have shown some advantages of dual quaternion filtering, but limited applicability for practical robotic applications. Besides, to the best knowledge of the authors, there is no dual quaternion-based 6-DoF SLAM framework so far, especially with a stereo setup.

Thus, we propose a feature-based stereo SLAM framework using a novel unscented filtering scheme with dual quaternions representing the 6-DoF ego-motions. More specifically, our main contributions are:

- A novel dual quaternion nonlinear filtering approach for 6-DoF pose estimation is proposed based on the unscented transform and a progressive Bayes update.
- A novel unscented transform-based approach is proposed for the map update.
- To ensure reliable feature observations in dynamic scenarios, an optical flow-based algorithm is proposed and embedded in the filtering scheme to exclude false data association.
- Based on the proposed tracking and mapping methods, the first dual quaternion-based stereo visual SLAM framework is developed.

Thorough evaluations in real-world scenarios are conducted using the KITTI [17] and EuRoC [18] datasets including indoor/outdoor and static/dynamic scenarios on diverse robotic platforms. The proposed framework gives robust and accurate performance comparable to state-of-the-art optimization-based visual SLAM. It is currently ranked the most accurate filtering-based SLAM system in the stereo setup category of KITTI[1].

The remainder of the paper is divided as follows: In Sec. II, preliminaries about dual quaternion pose parameterization and Bingham modeling are introduced. The proposed filtering technique is explained in Sec. III. Sec. IV presents the proposed SLAM framework with a detailed introduction for each function block. Results of real-world benchmarking are given in Sec. V and the work is concluded in Sec. VI.

II. Preliminaries

A. Pose Parameterization Using Dual Quaternions

In this work, 6-DoF poses are parameterized by dual quaternions [13] as

$$\mathbf{x} = \mathbf{x}_q + \frac{\epsilon}{2}\mathbf{t} \otimes \mathbf{x}_q \,, \tag{1}$$

with the unit quaternion $\mathbf{x}_q = \cos(\theta/2) + \mathbf{u}\sin(\theta/2)$ representing orientation and $\mathbf{t} \in \mathbb{R}^3$ the position. Here, the unit vector $\mathbf{u}$ denotes the rotation axis and θ the angle, such that a vector can be rotated accordingly. $\otimes$ denotes the Hamilton product [13] and ϵ is the dual unit ($\epsilon^2 = 0$). The arithmetic of dual quaternions is therefore the combination of the Hamilton product and dual number theory. Detailed introductions can be found in [19]. In this paper, we use the vector form

$$\mathbf{x} = [\mathbf{x}_q^\top, \mathbf{x}_p^\top]^\top \in \mathbb{R}^8 \,, \tag{2}$$

with $\mathbf{x}_p = \frac{1}{2}\mathbf{t} \otimes \mathbf{x}_q$. Thus, the pure translation can be computed as

$$\mathbf{t} = 2\,\mathbf{x}_p \otimes \mathbf{x}_q^{-1} \,, \tag{3}$$

with $\mathbf{x}_q^{-1}$ being the inverse of $\mathbf{x}_q$ [20]. The orientational term $\mathbf{x}_q$ inherently lies on the unit hypersphere $\mathbb{S}^3$.

B. Bingham Distribution

As introduced in [14], [21], the Bingham distribution on $\mathbb{S}^3$ is well suited for modeling uncertain unit quaternions as its dispersion is inherently antipodally symmetric, thus ensuring equal density at $\mathbf{x}_q$ and $-\mathbf{x}_q$, which denote the same orientation. It is defined as

$$f_{\mathcal{B}}(\mathbf{x}_q; \mathbf{M}, \mathbf{Z}) = \frac{1}{N(\mathbf{Z})} \exp(\mathbf{x}_q^\top \mathbf{M}\,\mathbf{Z}\,\mathbf{M}^\top \mathbf{x}_q) \,, \tag{4}$$

with $\mathbf{x}_q \in \mathbb{S}^3 \subset \mathbb{R}^4$. The orthonormal matrix $\mathbf{M}$ controls the rotation of its dispersion on $\mathbb{S}^3$, whereas the diagonal matrix $\mathbf{Z} = \mathrm{diag}(z_1, z_2, z_3, z_4)$ determines the concentration with $z_1 < z_2 < z_3 < z_4 \leq 0$. These two matrices can be generated from the eigendecomposition of a negative semi-definite matrix $\mathbf{C}_q$. In the remainder of the paper, we use $\mathbf{C}_q$ to denote the parameter matrix of the Bingham distribution.

III. Unscented Dual Quaternion Filtering

A. Prediction Using Unscented Transform

In this work, the rotational and translational components of dual quaternions are modeled by the Bingham and the Gaussian distribution respectively, i.e., $\mathbf{x}_q \sim \mathcal{B}(\mathbf{C}_q)$ and $\mathbf{t} \sim \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$. In [22], a deterministic sampling method was proposed for the Bingham distribution to draw sigma points and their weights preserving the first and second moment. We employ this method and further combine it with deterministic sampling for the Gaussian distribution modeling pure translation. The resulting unscented transform-based sampling approach is proposed as shown in Alg. 1, where dual quaternion samples are composed of the two components using Cartesian product.

The prediction step assumes the system dynamics model

$$\mathbf{x}_k = a(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) \otimes \mathbf{v}_k \,. \tag{5}$$

The time-invariant system noise

$$\mathbf{v}_k = [\mathbf{v}_{q,k}^\top, \mathbf{v}_{p,k}^\top]^\top \tag{6}$$

is modeled by $\mathbf{v}_{q,k} \sim \mathcal{B}(\mathbf{C}_q^v)$ and $\mathbf{v}_{t,k} \sim \mathcal{N}(\boldsymbol{\mu}_t^v, \boldsymbol{\Sigma}_t^v)$ with

$$\mathbf{v}_{p,k} = \frac{1}{2}\mathbf{v}_{t,k} \otimes \mathbf{v}_{q,k} \,. \tag{7}$$

[1]DQV-SLAM on http://www.cvlibs.net/datasets/kitti/eval_odometry.php

Algorithm 1 Sigma Point Generation

procedure sampleUT $(\mathbf{C}_q, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$

$\{(\boldsymbol{\sigma}_q^i, r^i)\}_{i=1:\alpha} \leftarrow$ sampleDet $(\mathbf{C}_q)$; // see [22]
$\{(\boldsymbol{\sigma}_t^j, s^j)\}_{j=1:\beta} \leftarrow$ sampleDet $(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$; // see [23]
$\{(\boldsymbol{\sigma}_x^k, w^k)\}_{k=1:(\alpha\cdot\beta)} \leftarrow \emptyset$, $k \leftarrow 1$;
for $i = 1$ to α **do**
  for $j = 1$ to β **do**
    $\boldsymbol{\sigma}_x^k \leftarrow [(\boldsymbol{\sigma}_q^i)^\top, (\frac{1}{2}\boldsymbol{\sigma}_t^j \otimes \boldsymbol{\sigma}_q^i)^\top]^\top$;
    $w^k \leftarrow r^i \cdot s^j$;
    $k \leftarrow k + 1$;
  end for
end for
return $\{(\boldsymbol{\sigma}_x^k, w^k)\}_{k=1:(\alpha\cdot\beta)}$

end procedure

System state samples drawn by Alg. 1 can thus be propagated through the system equation $a(\cdot,\cdot)$ together with the noise samples also drawn by Alg. 1, following the Cartesian product. The Bingham and Gaussian priors are then approximated from the resulting samples using the moment matching approach introduced in [14].

B. Progressive Bayes Update

For the update step, the pose prior is corrected given the measurement model

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{w}_k\,, \tag{8}$$

with $h(\cdot)$ denoting the observation function and $\mathbf{w}_k$ the measurement noise. The measurement noise is assumed to be additive, such that the likelihood function can be derived as

$$f(\mathbf{z}_k|\mathbf{x}_k) = f_{\mathbf{w}_k}(\mathbf{z}_k - h(\mathbf{x}_k)) \tag{9}$$

using Bayesian inference [20], [24], [25]. In order to adapt the sampling-based filtering scheme to the non-identity measurement model and to eliminate sample degeneration, we update the prior gradually with rescaled likelihood factors. We apply a progressive Bayesian inference scheme that has been used in a number of nonlinear filtering approaches [20], [26]. An introduction can be found in Alg. 2. During each progressive update step, we first draw deterministic samples according to Alg. 1, then calculate each sample's likelihood. The progressive step size is determined by ensuring the ratio between smallest and largest likelihood to be larger than a predefined threshold $\tau \in (0, 1]$ (Alg. 2, line 6-8). After reweighting the samples with the scaled likelihood, the posterior is approximated by moment matching [14]. We summarize the posterior parameters to be $\hat{\mathcal{C}}_x^k$ for convenience. Detailed theoretical foundation of this methodology can be found in [20], where it is applied for planar motion filtering.

IV. DQV-SLAM: Framework Description

Fig. 2 shows the proposed stereo SLAM framework. The system takes grayscale stereo images as input data and in a first step extracts ORB features [27] that are used as measurement input in the following. ORB features are fast to extract and match while providing similar performance to the popular SIFT and SURF features which makes them a good choice for feature-based SLAM algorithms [1], [27].

Algorithm 2 Nonlinear Progressive Update

procedure progressiveUpdate $(\mathbf{C}_q^{k|k-1}, \boldsymbol{\mu}_t^{k|k-1}, \boldsymbol{\Sigma}_t^{k|k-1}, \mathbf{z}_k)$

1: $\Lambda \leftarrow 1$, $i \leftarrow 1$;
2: $\{\hat{\mathbf{C}}_q^k, \hat{\boldsymbol{\mu}}_t^k, \hat{\boldsymbol{\Sigma}}_t^k\} \leftarrow \{\mathbf{C}_q^{k|k-1}, \boldsymbol{\mu}_t^{k|k-1}, \boldsymbol{\Sigma}_t^{k|k-1}\}$;
3: **while** $\Lambda > 0$ **do**
4: $\quad\{(\boldsymbol{\sigma}_x^j, w^j)\}_{j=1:\alpha} \leftarrow$ sampleUT $(\hat{\mathbf{C}}_q^k, \hat{\boldsymbol{\mu}}_t^k, \hat{\boldsymbol{\Sigma}}_t^k)$;
5: $\quad\{s^j\}_{j=1:\alpha} \leftarrow$ likelihood $(\{\boldsymbol{\sigma}_x^j\}_{j=1:\alpha}, \mathbf{z}_k)$;
6: $\quad s_{\min} \leftarrow \min(\{s^j\}_{j=1:\alpha})$;
7: $\quad s_{\max} \leftarrow \max(\{s^j\}_{j=1:\alpha})$;
8: $\quad\lambda_i \leftarrow \min(\Lambda, \frac{\log(\tau)}{\log(s_{\min}/s_{\max})})$;
9: $\quad$**for** $j = 1$ **to** α **do**
10: $\quad\quad w^j \leftarrow (s^j)^{\lambda_i} \cdot w^j$;
11: $\quad$**end for**
12: $\quad\left\{\hat{\mathbf{C}}_q^k, \hat{\boldsymbol{\mu}}_t^k, \hat{\boldsymbol{\Sigma}}_t^k\right\} \leftarrow$ momentMatch $(\{(\boldsymbol{\sigma}_x^j, w^j)\}_{j=1:\alpha})$;
13: $\quad i \leftarrow i + 1$, $\Lambda \leftarrow \Lambda - \lambda_i$;
14: **end while**
15: $\hat{\mathcal{C}}_x^k \leftarrow \{\hat{\mathbf{C}}_q^k, \hat{\boldsymbol{\mu}}_t^k, \hat{\boldsymbol{\Sigma}}_t^k\}$;
16: **return** $\hat{\mathcal{C}}_x^k$

end procedure

The unscented dual quaternion filter from Sec. III is adapted for the stereo SLAM setup and further incorporates a probabilistic mapping scheme that incrementally builds a global map $\mathcal{M}_k$ as a point cloud of landmarks from the observed ORB features.

$$\mathcal{M}_k = \{\mathbf{l}_k^i\}_{i=1:n}\,,\ \mathbf{l}_k^i \in \mathbb{R}^3\,. \tag{10}$$

Each landmark also carries appearance information encoded by an ORB-descriptor, which is used for data association.

The filter state for the SLAM framework consists of two parts: the dual quaternion camera pose $\mathbf{x}_k$ and the map $\mathcal{M}_k$. Uncertainty of the pose is modeled by the combined Bingham and Gaussian parameters $\hat{\mathcal{C}}_x^k$ (cf Sec. III-B) and an individual covariance matrix $\mathbf{P}_k^i$ is kept per landmark $\mathbf{l}_k^i$.

The filter recursion after the preprocessing stage consists of four steps *pose prediction*, *data association*, *pose update* and *map update* which will be explained in Sections IV-A to IV-E. In case of a successful loop closure detections, the update steps are replaced by *pose graph optimization* and a subsequent *map correction* step. The loop closure module is detailed in Section IV-F.

A. System Model and Prediction

The system dynamics model is assumed as

$$\mathbf{x}_{k|k-1} = \mathbf{x}_{k-1} \otimes \mathbf{u}_k \otimes \mathbf{v}_k\,, \tag{11}$$

where the system input $\mathbf{u}_k$ is computed as the dual quaternion transformation between consecutive frames using the visual odometry library *libviso2* [28]. The pose prior $\mathbf{x}_{k|k-1}$ can then be predicted based on Sec. III-A. Using visual odometry as system input for visual SLAM is shown by [29], [30] to

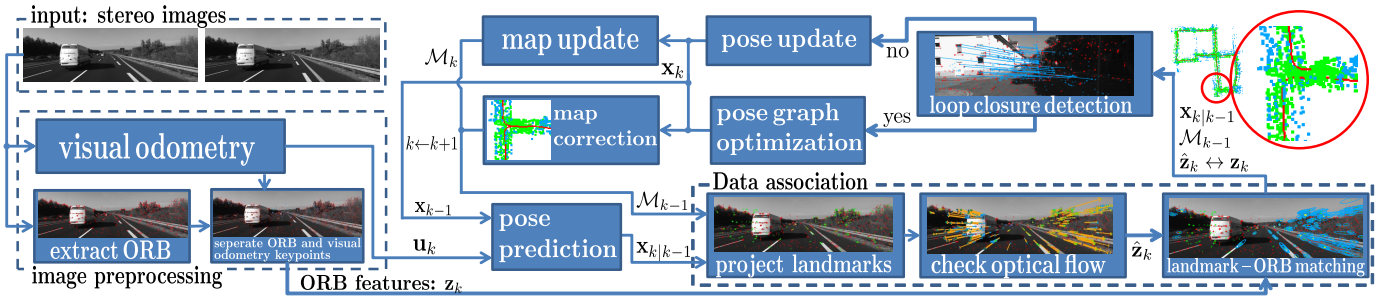


Fig. 2: Overview of the proposed stereo visual SLAM framework.

be advantageous over generic motion models, e.g., constant velocity, which cannot properly handle sudden movements.

However, as *libviso2* extracts features from regions with large image gradient [28], similar to the ORB keypoint detector, it is essential to assure that the features extracted in the prediction step are not correlated to the ORB-features used in the update step. Inspired by [30], we exclude observed ORB-features that are close to the features used during prediction from being associated during the update step (see image preprocessing in Fig. 2).

B. Measurement Model: Landmark Projection

The pose prior $\mathbf{x}_{k|k-1}$ is corrected progressively using the observed ORB-features $\mathbf{z}_k$ that are associated with landmarks $\mathbf{l}_{k-1}$ from the map $\mathcal{M}_{k-1}$. The measurements are by convention represented by the stereo pixel coordinates of the features after image rectification, namely

$$\mathbf{z}_k = [\, u_l, v_l, d \,]^\top \in \mathbb{R}^3 \,, \tag{12}$$

with $[\, u_l, v_l \,]^\top$ being the left pixel coordinates and $d = u_l - u_r$ the corresponding disparity across the image pair.

To calculate the expected measurements $\hat{\mathbf{z}}_k^i$, the landmarks $\mathbf{l}_{k-1}^i$ are projected onto the image plane with respect to the pose prior $\mathbf{x}_{k|k-1}$. They are then associated with the current observations, for which their projected uncertainty is required. In [31], sigma point-based methods are proven to give good approximations when propagating uncertainty through the nonlinear camera model. Inspired by this, we propose a landmark projection approach based on the unscented transform illustrated in Fig. 3 and explained in Alg. 3. Each landmark $\mathbf{l}_{k-1}^i$ in the current field of view is projected onto the image plane together with its uncertainty $\mathbf{P}_{k-1}^i$ using sigma points. Besides, a time-invariant observation pixel noise $\mathbf{R}$ is added to the projected covariance and finally gives the measurement noise $\mathbf{S}_k^i$. Using the observation noise $\mathbf{R}$ as an additive term, instead of augmenting the landmark state, is shown in [12] to only have a negligible loss in accuracy for low-dimensional states (e.g. the 3D landmark positions) while reducing the number of sigma points and thus the computational cost.

C. Data Association: Optical Flow-based Feature Matching

In the data association stage, the projected landmarks must be reliably associated with the currently observed ORB-features. However, this can be risky in dynamic scenarios

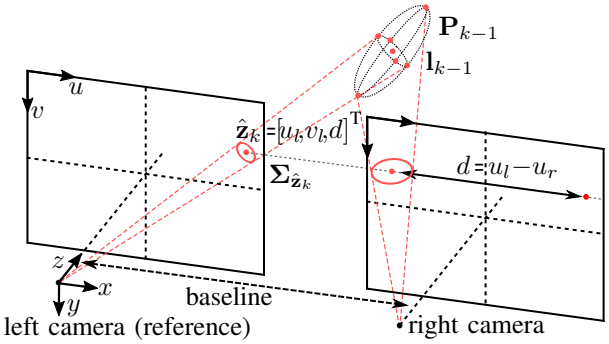


Fig. 3: Stereo landmark projection. Landmark $\mathbf{l}_{k-1}$ with uncertainty $\mathbf{P}_{k-1}$ is projected onto the image plane using sigma points, resulting in an expected observation $\hat{\mathbf{z}}_k$ and covariance $\boldsymbol{\Sigma}_{\hat{\mathbf{z}}_k}$.

Algorithm 3 Landmark Projection

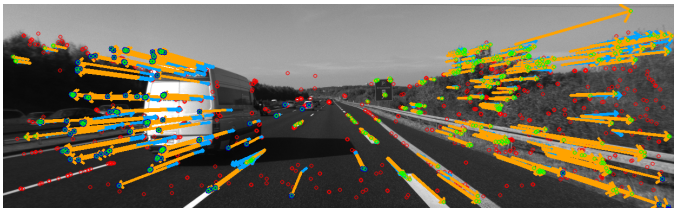
procedure projLandmark $(\mathbf{x}_{k|k-1}, \mathbf{l}_{k-1}, \mathbf{P}_{k-1}, \mathbf{R})$

1: $\{(\boldsymbol{\sigma}_{l,k-1}^j, w^j)\}_{j=1:\beta} \leftarrow$ sampleDet $(\mathbf{l}_{k-1}, \mathbf{P}_{k-1})$;
2: $\{\hat{\mathbf{z}}_k^j\}_{j=1:\beta} \leftarrow$ proj $(\mathbf{x}_{k|k-1}, \{\boldsymbol{\sigma}_{l,k-1}^j\}_{j=1:\beta})$;
3: $\hat{\mathbf{z}}_k \leftarrow \sum_{j=1}^{\beta} w^j \cdot \hat{\mathbf{z}}_k^j$;
4: $\boldsymbol{\Sigma}_{\hat{\mathbf{z}}_k} \leftarrow \sum_{j=1}^{\beta} w^j (\hat{\mathbf{z}}_k^j - \hat{\mathbf{z}}_k)(\hat{\mathbf{z}}_k^j - \hat{\mathbf{z}}_k)^\top$;
5: $\mathbf{S}_k \leftarrow \boldsymbol{\Sigma}_{\hat{\mathbf{z}}_k} + \mathbf{R}$;
6: **return** $\{\hat{\mathbf{z}}_k, \mathbf{S}_k\}$

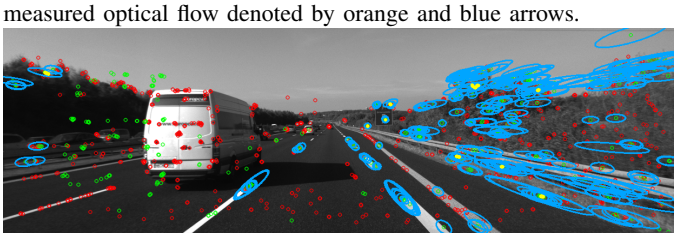
end procedure

as shown in Fig. 4, where some features are extracted from dynamic surroundings, e.g., the moving vehicles. In order to eliminate this issue, an optical flow-based approach is proposed for feature matching.

For each landmark $\mathbf{l}_{k-1}^i$ about to be associated with the observation, we first compute its predicted optical flow as the translation between its projection onto the previous and current image, for which the last pose posterior and the current prior are used, respectively. The predicted optical flow is shown by the orange arrows in Fig. 4(a). Meanwhile, its observed optical flow (blue arrow) is calculated using the *OpenCV* library [32]. Landmarks on the static environment will have almost identical directions and magnitude of predicted and observed optical flow. In contrast, landmarks projected on dynamic objects will be observed to have optical flow that largely differs from the predicted direction or magnitude. These landmark projections are excluded from being further associated (green circles without ellipses in Fig 4(b)).



(a) ORB features (red), projected landmarks (green), predicted and measured optical flow denoted by orange and blue arrows.



(b) Covariance ellipses defining the geometric search region (blue) and successfully associated landmarks (yellow).

Fig. 4: Data association based on optical flow direction and feature matching using Mahalanobis and descriptor distance.

The remaining projected landmarks $\hat{\mathbf{z}}_k$ are then associated with the observed features based on the Mahalanobis distance given by the projected covariance $\mathbf{S}_k$ from Alg. 3 (shown as blue ellipses in Fig. 4(b)). For features inside this search region, appearance-based matching is done in the ORB descriptor space to find the most similar-looking pairs. Successful associations are shown as yellow lines in Fig. 4(b). The aforementioned correspondence search is first done for left and right image separately. Afterwards, a coherence check is performed to ensure that the associations found in both images satisfy the epipolar constraint. In the end, the data association step gives a set of m associated landmarks and features $\{(\mathbf{l}_{k-1}^j, \mathbf{z}_k^j)\}_{j=1:m}$ that will be used for pose and map update.

D. Likelihood and Pose Update

We assume that the measurement noise is independently Gaussian-distributed for individual features. The likelihood in Alg. 2 line 5 can then be computed as the product of the likelihood values given by the predicted measurements $\hat{\mathbf{z}}_k^j$ and their associated observations $\mathbf{z}_k^j$. This can further be derived as

$$s = \prod_{j=1}^{m} \frac{1}{N_j} \exp\left(-\frac{1}{2}(\hat{\mathbf{z}}_k^j - \mathbf{z}_k^j)^\top (\mathbf{S}_k^j)^{-1}(\hat{\mathbf{z}}_k^j - \mathbf{z}_k^j)\right), \quad (13)$$

with $N_j = (2\pi)^{\frac{3}{2}} \sqrt{\det(\mathbf{S}_k^j)}$ being each Gaussian normalization constant. The projected landmark $\hat{\mathbf{z}}_k^j$ and its covariance $\mathbf{S}_k^j$ are calculated by Alg. 3. Given the likelihood (13), the pose prior is updated according to Sec. III-B. To avoid numerical instabilities in the progressive update, the maximum number of correspondences processed at once must be limited. When more than $M_{\max} = 30$ valid correspondences are found, several subsets are processed sequentially while adding a regularization term to the parameter matrix $\hat{\mathcal{C}}_x^k$ between the consecutive rounds of progressive update. The subsets are determined such that the correspondences are evenly distributed over the image.

Algorithm 4 Landmark Update

procedure updateLandmark $(\hat{\mathcal{C}}_x^k, \mathbf{l}_{k-1}, \mathbf{P}_{k-1}, \mathbf{z}_k, \mathbf{R})$

1: $\{\hat{\mathbf{C}}_q^k, \hat{\boldsymbol{\mu}}_t^k, \hat{\boldsymbol{\Sigma}}_t^k\} \leftarrow \hat{\mathcal{C}}_x^k$;
2: $\{(\hat{\boldsymbol{\sigma}}_{x,k}^i, w^i)\}_{i=1:\alpha} \leftarrow$ sampleUT $(\hat{\mathbf{C}}_q^k, \hat{\boldsymbol{\mu}}_t^k, \hat{\boldsymbol{\Sigma}}_t^k)$;
3: $\{(\boldsymbol{\sigma}_{l,k-1}^j, p^j)\}_{j=1:\beta} \leftarrow$ sampleDet $(\mathbf{l}_{k-1}, \mathbf{P}_{k-1})$;
4: $\Delta \leftarrow \alpha \cdot \beta$;
5: $\{\hat{\mathbf{z}}_k^\delta\}_{\delta=1:\Delta}, \{w_{\hat{z}}^\delta\}_{\delta=1:\Delta}, \{\mathbf{l}_{\text{rep}}^\delta\}_{1:\Delta} \leftarrow \emptyset$;
6: **for** $i = 1$ to α **do**
7: $\quad \{\hat{\mathbf{z}}_k^\delta\}_{\delta=\beta(i-1)+1:\beta i} \leftarrow$ proj $(\hat{\boldsymbol{\sigma}}_{x,k}^i, \{\boldsymbol{\sigma}_{l,k-1}^j\}_{j=1:\beta})$;
8: $\quad \{w_{\hat{z}}^\delta\}_{\delta=\beta(i-1)+1:\beta i} \leftarrow$ multiply $(w^i, \{p^j\}_{j=1:\beta})$;
9: $\quad \{\mathbf{l}_{\text{rep}}^\delta\}_{\delta=\beta(i-1)+1:\beta i} \leftarrow \{\boldsymbol{\sigma}_{l,k-1}^j\}_{j=1:\beta}$;
10: **end for**
11: $\hat{\mathbf{z}}_k \leftarrow \sum_{\delta=1}^{\Delta} w_{\hat{z}}^\delta \cdot \hat{\mathbf{z}}_k^\delta$;
12: $\mathbf{S}_k \leftarrow \sum_{\delta=1}^{\Delta} w_{\hat{z}}^\delta (\hat{\mathbf{z}}_k^\delta - \hat{\mathbf{z}}_k)(\hat{\mathbf{z}}_k^\delta - \hat{\mathbf{z}}_k)^\top + \mathbf{R}$;
13: $\mathbf{S}_k^{lz} \leftarrow \sum_{\delta=1}^{\Delta} w_{\hat{z}}^\delta (\mathbf{l}_{\text{rep}}^\delta - \mathbf{l}_{k-1})(\hat{\mathbf{z}}_k^\delta - \hat{\mathbf{z}}_k)^\top$;
14: $\mathbf{K}_k = \mathbf{S}_k^{lz}\, \mathbf{S}_k^{-1}$;
15: $\mathbf{l}_k = \mathbf{l}_{k-1} + \mathbf{K}_k(\mathbf{z}_k - \hat{\mathbf{z}}_k)$;
16: $\mathbf{P}_k = \mathbf{P}_{k-1} - \mathbf{K}_k\, \mathbf{S}_k\, \mathbf{K}_k^T$;
17: **return** $\{\mathbf{l}_k, \mathbf{P}_k\}$;

end procedure

E. Map Update

A static map is assumed in our framework. Each landmark position $\mathbf{l}_{k-1} \in \mathbb{R}^3$ is assumed to be independently distributed and have zero-centered Gaussian noise denoted by $\mathbf{P}_{k-1}$. For large-scale mapping, explicit modeling of the cross-correlation between landmarks is computationally infeasible (cf. FastSLAM [33]). Therefore, landmarks are updated individually based on the unscented transform as shown in Alg. 4. One single map is maintained in the proposed system and the cross-correlation between pose and landmarks is incorporated in a sample-based manner.

For each re-observed landmark, sigma points are drawn from the current pose posterior $\hat{\mathcal{C}}_x^k$ and the last landmark posterior $\{\mathbf{l}_{k-1}, \mathbf{P}_{k-1}\}$, respectively. Given each pose sample, the whole set of landmark sigma points is projected onto the image plane to get the predicted measurements. Their weights are computed by multiplying the weights of landmark and pose samples. The predicted measurements, their weights and the landmark samples are then concatenated each individually (Alg. 4, line 7-9). The remaining steps of the algorithm follow the UKF update without state augmentation as proposed in [12].

For newly observed features, new landmarks will be added to the map and their uncertainty $\mathbf{P}_k$ is initialized by projecting sigma points from the observation noise $\mathbf{R}$ to global coordinates, which can be seen as the inverse operation of Alg. 3. To only keep trackable landmarks in the map, we impose a culling policy that removes landmarks which are not re-observed within 5 frames.

When initializing landmarks from stereo observations, we differentiate between close and far points similar to [1] and [9]. As suggested in [9], landmarks within a distance of 40 times the baseline are represented as 3D positions, while far

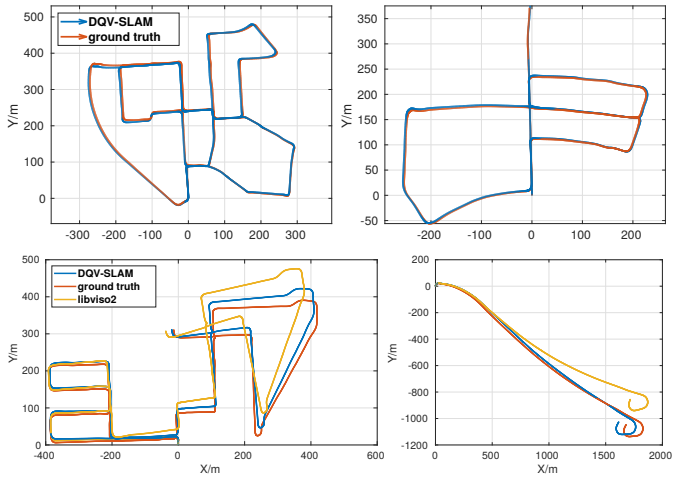


Fig. 5: Sample trajectories given by the proposed SLAM system for the KITTI dataset. Top: Seq. 00, 05. Bottom: Seq. 08, 01 (open loop).

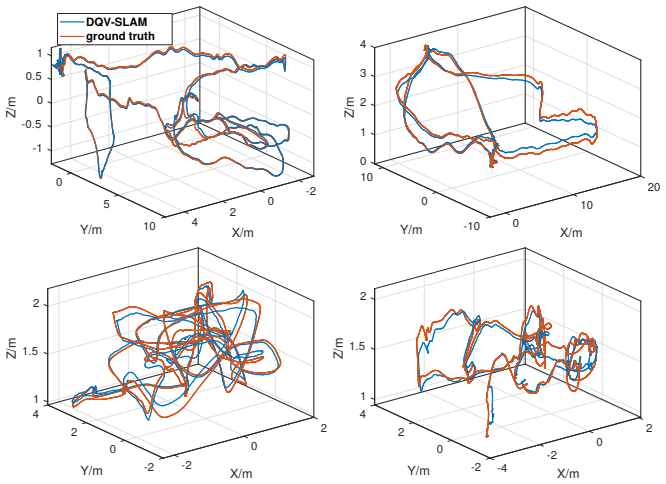


Fig. 6: Sample trajectories given by the proposed SLAM system for the EuRoC dataset. Top: MH-01, MH-05. Bottom: V1-02, V2-01.

points are parameterized by the inverse depth representation introduced in [34].

F. Loop Closure and Pose Graph Optimization

As mentioned in most SLAM systems, accumulated odometry errors, also known as drift, should be corrected once a loop closure is detected. In our proposed framework, loop closure detection is activated if landmarks re-enter the current field of view after not having been observed for more than a certain number of frames. Though the drift initially cannot be estimated, we assume that it is limited such that previously observed landmarks should be located near the current field of view at a loop closure point. After associating these landmarks with current feature observations, a global pose estimate is calculated using the Perspective-n-Point (PnP) method with an embedded RANSAC scheme for eliminating outliers [32]. Once similar global pose estimates are given in three consecutive frames with a sufficient percentage of inliers in each RANSAC process, the loop closure is finally confirmed.

The pose estimates from the unscented dual quaternion filter are inserted into a pose graph as odometry constraints. Once a loop closure is detected, an edge closing the graph with the measured drift given from the PnP method is added and the pose graph optimization is triggered. The optimization is done by the *gtsam* framework [35]. After that, the filter is reset according to the optimized states. For correcting the map, each landmark stores the index of its last observing pose and gets corrected correspondingly.

V. Evaluation

The proposed SLAM framework is implemented in C++ and evaluated using the KITTI [17] and EuRoC [18] datasets for stereo-only setup and compared to other state-of-the-art SLAM systems. The KITTI dataset contains outdoor driving scenarios of both urban and highway surroundings, whereas the EuRoC dataset was recorded indoor onboard a flying Micro Aerial Vehicle (MAV). For the two datasets, we employ their respective standard evaluation metric given in [17] and [36]. The system is configured to extract 1000 ORB-features per image and use landmarks within a distance of 140 times the stereo baseline.

A. Tracking Accuracy

Fig. 5 shows four sample trajectories from the KITTI dataset. Our system gives very accurate results on the 00 and 05 sequences (upper row), where several loop closure points occur. In the bottom row, two open loop trajectories are shown, where no loop closure occurs. Here, the trajectory is estimated only by the filtering algorithm and no pose graph optimization is performed. We compare our system to *libviso2* which gives the system input for the prediction step. It is apparent that the proposed progressive update scheme using mapped landmarks gives significant improvement compared to the pure odometry where only correspondences between consecutive frames are used. Although drift cannot be corrected for the open loop trajectories, the result of our system is still close to the ground truth. Our overall translational error is 1.11% for the training set and 1.47% for the test set. This increased error mainly results from the challenging highway sequences (20 and 21) where textureless images occur more frequently. In Table I, we give detailed benchmarking on the KITTI training set. For comparison, several state-of-the-art SLAM systems are used, including SuMa [37] (LiDAR SLAM), ProSLAM [38] and ORB-SLAM2 [1] (feature-based using keyframes). All the compared systems are optimization-based. For all the sequences, the proposed progressive update is shown to give significant improvement over *libviso2*. Regarding translational error, the proposed DQV-SLAM performs equally or better compared to SuMa and ProSLAM for 3 of the 11 sequences.

For the EuRoC dataset, we show four sample trajectories in Fig. 6 from both the machine hall (top row) and Vicon room (bottom row) environments. The proposed SLAM system gives accurate results also in the challenging MAV scenarios that comprise fast motion and varying lighting conditions. Table II further shows detailed benchmarking results with other state-of-the-art systems. For the full SLAM configuration, we

TABLE I: Relative error t_{rel} (%) / r_{rel} (°/100m) on KITTI odometry dataset. * denotes no loop closure. Bold numbers denote the best.

Approach	Sequence										
	00	01*	02	03*	04*	05	06	07	08*	09	10*
ORB-SLAM2 [1]	0.7/0.3	1.4/0.2	0.8/0.2	0.7/0.2	0.5/0.1	0.4/0.2	0.5/0.2	0.5/0.3	1.1/0.3	0.9/0.3	0.6/0.3
DQV-SLAM	**0.7**/0.4	3.5/0.7	**1.0**/0.3	1.4/0.4	1.0/0.4	0.8/0.3	1.1/0.4	0.6/0.4	1.3/0.4	1.6/0.6	0.9/0.5
libviso2 [28]	2.7/1.2	4.3/1.0	2.2/0.9	2.3/1.0	1.1/0.8	2.2/1.2	1.3/0.9	2.4/1.8	2.8/1.3	2.8/1.2	1.4/1.2
SuMa [37]	**0.7**/0.2	**1.7**/0.5	1.2/0.4	0.7/0.5	**0.4**/0.3	**0.4**/0.2	**0.5**/0.3	0.7/0.6	1.2/0.4	**0.6**/0.2	0.7/0.3
ProSLAM [38]	**0.7**/0.3	**1.7**/0.5	**1.0**/0.3	**0.6**/0.2	0.5/0.5	0.8/0.3	**0.5**/0.3	**0.5**/0.3	**1.0**/0.3	1.0/0.3	**0.5**/0.2

TABLE II: ATE (Absolute Trajectory Error) of RMSE in meter on EuRoC. 'x' denotes system failing. Bold numbers denote the best.

Approach	Sequence										
	V1-1	V1-2	V1-3	V2-1	V2-2	V2-3	MH-1	MH-2	MH-3	MH-4	MH-5
ORB-SLAM2 [1]	0.04	0.02	0.05	0.04	0.04	x	0.04	0.02	0.03	0.12	0.06
DQV-SLAM	0.10	0.08	0.15	0.19	0.20	x	**0.06**	**0.09**	**0.13**	0.35	**0.19**
VI-SLAM [39]	**0.04**	**0.05**	**0.11**	**0.10**	**0.18**	x	0.11	**0.09**	0.19	**0.27**	0.23
DQV-SLAM no LC	0.30	0.29	0.37	0.45	0.48	x	0.25	0.27	0.40	0.82	0.53
S-MSCKF [10]	0.23	0.20	0.30	0.14	0.31	x	n/a	n/a	0.37	0.48	0.48
ROVIO [7], results from [10]	0.22	0.21	0.34	0.28	0.21	0.26	n/a	n/a	0.44	1.34	0.99

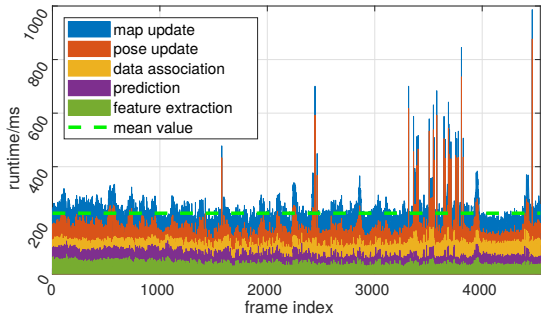


Fig. 7: Runtime KITTI 00.

compare to VI-SLAM [39], a keyframe-based stereo visual-inertial SLAM system. Our system is more or equally accurate as VI-SLAM for four of the five machine hall sequences and performs competitively for the Vicon room, despite only using camera input and no inertial data. The bottom part of Table II compares the proposed system with loop closure disabled to two filtering-based visual-inertial odometry approaches, ROVIO [7] (monocular) and S-MSCKF [10] (stereo). For the machine hall sequences which contain features at larger depth, DQVSLAM is more accurate than ROVIO. It stays comparable to the VIO systems for the remaining sequences although it does not process higher-order motion cues from an IMU.

ORB-SLAM2, which is among the best visual SLAM systems, is used as a baseline for our benchmarking (not included into comparison). Generally speaking, our system performs competitively robust and accurate compared to the other state-of-the-art visual SLAM systems in challenging indoor and outdoor real-world scenarios.

B. Timing

The proposed system shows an averaged processing time of 220 ms per frame over the whole KITTI dataset on a mobile computer (Intel Core i5-7200U CPU, 8GB RAM). Fig. 7 shows the timing result for KITTI 00 among different processing stages. This sequence contains several loop closure points, which can be observed as peaks when pose graph optimizations are performed. The overall processing frame rate is near 5 Hz. It is in the same order of magnitude as the input frame rate (10 Hz), yet not real-time, due to the sampling-approximation-based filtering scheme and unoptimized code for prototyping.

VI. Conclusion

In this paper, we presented DQV-SLAM, a novel feature-based stereo visual SLAM framework using an unscented dual quaternion filtering and mapping approach. A novel implementation of the 6-DoF SLAM problem using the dual quaternion formalism and methods from directional estimation is given. An optical flow-based approach is embedded into the pose filter to ensure robust data association in dynamic scenarios. The system is able to reliably detect loop closure and correct drift. It is ranked the best filtering-based approach in the visual odometry/SLAM category of KITTI. For different real-world scenarios, the proposed framework shows competitive performance compared to state-of-the-art systems. However, there is still much potential that can be exploited. The current unscented transform-based sampling scheme provides a sparse coverage of the state space and samples are only placed on the principal axes. With the employed measurement model, this is problematic, as small changes in orientation of the camera can have large influence on the projected landmark positions. To provide for a better coverage of the manifold of orientations, sampling schemes that draw a larger number of more evenly distributed samples could be employed [40], [41]. The proposed filtering scheme can also be improved by considering the correlation between the orientation and translation part [42]. Furthermore, the efficiency should be improved and pose graph optimization be moved to a parallel thread to enable future deployment on real-time robotic platforms. Besides, other kinds of sensors, e.g., LiDAR or IMU, can be incorporated into the framework using sensor fusion techniques for better performance.

Acknowledgments

This work is supported by the German Research Foundation (DFG) under grant HA 3789/16-1.

REFERENCES

[1] R. Mur-Artal and J. D. Tards, "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, Oct 2017.

[2] T. Pire, T. Fischer, G. Castro, P. De Cristóforis, J. Civera, and J. J. Berlles, "S-PTAM: Stereo Parallel Tracking and Mapping," *Robotics and Autonomous Systems*, vol. 93, pp. 27–42, 2017.

[3] J. Engel, J. Stckler, and D. Cremers, "Large-Scale Direct SLAM with Stereo Cameras," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2015)*, Sept 2015, pp. 1935–1942.

[4] R. Wang, M. Schwörer, and D. Cremers, "Stereo DSO: Large-Scale Direct Sparse Visual Odometry with Stereo Cameras," in *International Conference on Computer Vision (ICCV 2017), Venice, Italy*, 2017.

[5] H. Strasdat, J. M. Montiel, and A. J. Davison, "Visual SLAM: Why Filter?" *Image and Vision Computing*, vol. 30, no. 2, pp. 65–77, 2012.

[6] M. Karrer, M. Agarwal, M. Kamel, R. Siegwart, and M. Chli, "Collaborative 6DoF Relative Pose Estimation for two UAVs with Overlapping Fields of View," in *IEEE International Conference on Robotics and Automation (ICRA 2018)*, 2018.

[7] M. Bloesch, M. Burri, S. Omari, M. Hutter, and R. Siegwart, "Iterated Extended Kalman Filter Based Visual-Inertial Odometry using Direct Photometric Feedback," *The International Journal of Robotics Research*, vol. 36, no. 10, pp. 1053–1072, 2017.

[8] R. Sim, P. Elinas, and J. J. Little, "A Study of the Rao-Blackwellised Particle Filter for Efficient and Accurate Vision-Based SLAM," *International Journal of Computer Vision*, vol. 74, no. 3, pp. 303–318, Sep 2007.

[9] L. M. Paz, P. Pinis, J. D. Tards, and J. Neira, "Large-Scale 6-DOF SLAM With Stereo-in-Hand," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 946–957, Oct 2008.

[10] K. Sun, K. Mohta, B. Pfrommer, M. Watterson, S. Liu, Y. Mulgaonkar, C. J. Taylor, and V. Kumar, "Robust Stereo Visual Inertial Odometry for Fast Autonomous Flight," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 965–972, April 2018.

[11] A. I. Mourikis and S. I. Roumeliotis, "A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, April 2007, pp. 3565–3572.

[12] C. Kim, R. Sakthivel, and W. K. Chung, "Unscented FastSLAM: A Robust and Efficient Solution to the SLAM Problem," *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 808–820, Aug 2008.

[13] W. R. Hamilton, "On Quaternions; or on a New System of Imaginaries in Algebra," *Philosophical Magazine Series 3*, vol. 25, no. 163, pp. 10–13, 1844.

[14] I. Gilitschenski, G. Kurz, S. J. Julier, and U. D. Hanebeck, "Unscented Orientation Estimation Based on the Bingham Distribution," *IEEE Transactions on Automatic Control*, vol. 61, no. 1, pp. 172–177, Jan 2016.

[15] I. Gilitschenski, G. Kurz, and U. D. Hanebeck, "A Stochastic Filter for Planar Rigid-Body Motions," in *2015 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2015)*, Sept 2015, pp. 13–18.

[16] R. A. Srivatsan, G. T. Rosen, D. F. N. Mohamed, and H. Choset, "Estimating SE(3) Elements Using a Dual Quaternion Based Linear Kalman Filter," in *Robotics: Science and Systems*, 2016.

[17] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite," in *Conference on Computer Vision and Pattern Recognition (CVPR 2012)*, 2012.

[18] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The EuRoC Micro Aerial Vehicle Datasets," *The International Journal of Robotics Research*, 2016.

[19] K. Li, G. Kurz, L. Bernreiter, and U. D. Hanebeck, "Simultaneous Localization and Mapping Using a Novel Dual Quaternion Particle Filter," in *21st International Conference on Information Fusion (Fusion 2018)*. Cambridge, United Kingdom: IEEE, July 2018.

[20] ——, "Nonlinear Progressive Filtering for SE(2) Estimation," in *21th International Conference on Information Fusion (Fusion 2018)*. Cambridge, United Kingdom: IEEE, July 2018.

[22] I. Gilitschenski, G. Kurz, S. J. Julier, and U. D. Hanebeck, "Efficient Bingham filtering based on saddlepoint approximations," in *2014 International Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI 2014)*, Sept 2014, pp. 1–7.

[21] R. A. Srivatsan, M. Xu, N. Zevallos, and H. Choset, "Probabilistic Pose Estimation Using a Bingham Distribution-based Linear Filter," *The International Journal of Robotics Research*, June 2018.

[23] E. A. Wan and R. Van Der Merwe, "The Unscented Kalman Filter for Nonlinear Estimation," in *The IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC.*, 2000, pp. 153–158.

[24] F. Pfaff, K. Li, and U. D. Hanebeck, "Fourier filters, grid filters, and the fourier-interpreted grid filter," in *Proceedings of the 22nd International Conference on Information Fusion (Fusion 2019)*, Ottawa, Canada, July 2019.

[25] ——, "Association likelihoods for directional estimation," in *Proceedings of the 2019 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2019)*, Taipei, Republic of China, May 2019.

[26] J. Steinbring and U. D. Hanebeck, "Progressive Gaussian Filtering Using Explicit Likelihoods," in *IEEE 17th International Conference on Information Fusion (FUSION 2014), 2014*, 2014, pp. 1–8.

[27] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An Efficient Alternative to SIFT or SURF," in *2011 International Conference on Computer Vision*, Nov 2011, pp. 2564–2571.

[28] A. Geiger, J. Ziegler, and C. Stiller, "Stereoscan: Dense 3D Reconstruction in Real-Time," in *2011 IEEE Intelligent Vehicles Symposium (IV 2011)*, 2011, pp. 963–968.

[29] P. F. Alcantarilla, L. M. Bergasa, and F. Dellaert, "Visual Odometry Priors for Robust EKF-SLAM," in *2010 IEEE International Conference on Robotics and Automation*, May 2010, pp. 3501–3506.

[30] J. Kim, K.-J. Yoon, and I. S. Kweon, "Robust 3-D Visual SLAM in a Large-Scale Environment," in *Robotics Research*, C. Pradalier, R. Siegwart, and G. Hirzinger, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 519–534.

[31] T. D. Barfoot and P. T. Furgale, "Associating Uncertainty with Three-Dimensional Poses for Use in Estimation Problems," *IEEE Transactions on Robotics*, vol. 30, no. 3, pp. 679–693, 2014.

[32] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[33] M. Montemerlo, S. Thrun, D. Koller, B. Wegbreit, *et al.*, "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem," *AAAI/IAAI*, vol. 593598, 2002.

[34] J. Civera, A. J. Davison, and J. M. M. Montiel, "Inverse Depth Parametrization for Monocular SLAM," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 932–945, Oct 2008.

[35] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, "iSAM2: Incremental Smoothing and Mapping Using the Bayes Tree," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 216–235, 2012.

[36] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A Benchmark for the Evaluation of RGB-D SLAM Systems," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2012, pp. 573–580.

[37] J. Behley and C. Stachniss, "Efficient Surfel-Based SLAM using 3D Laser Range Data in Urban Environments," in *Robotics: Science and Systems (RSS 2018)*, 2018.

[38] D. Schlegel, M. Colosi, and G. Grisetti, "ProSLAM: Graph SLAM from a Programmer's Perspective," *arXiv preprint arXiv:1709.04377*, 2017.

[39] A. Kasyanov, F. Engelmann, J. Stückler, and B. Leibe, "Keyframe-based Visual-Inertial Online SLAM with Relocalization," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2017)*. IEEE, 2017, pp. 6662–6669.

[40] K. Li, D. Frisch, B. Noack, and U. D. Hanebeck, "Geometry-Driven Deterministic Sampling for Nonlinear Bingham Filtering," in *Proceedings of the 2019 European Control Conference (ECC 2019)*, Naples, Italy, June 2019.

[41] K. Li, F. Pfaff, and U. D. Hanebeck, "Hyperspherical Deterministic Sampling Based on Riemannian Geometry for Improved Nonlinear Bingham Filtering," in *Proceedings of the 22nd International Conference on Information Fusion (Fusion 2019)*, June 2019.

[42] ——, "Geometry-Driven Stochastic Modeling of SE(3) States Based on Dual Quaternion Representation," in *Proceedings of the 2019 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2019)*, Taipei, Republic of China, May 2019.