2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

AC/DCC : Accurate Calibration of Dynamic Camera Clusters for Visual SLAM

Jason Rebello* Angus Fung§ and Steven L. Waslander†

***Abstract*— In order to relate information across cameras in a Dynamic Camera Cluster (DCC), an accurate time-varying set of extrinsic calibration transformations need to be determined. Previous calibration approaches rely solely on collecting measurements from a known fiducial target which limits calibration accuracy as insufficient excitation of the gimbal is achieved. In this paper, we improve DCC calibration accuracy by collecting measurements over the entire configuration space of the gimbal and achieve a 10X improvement in pixel re-projection error. We perform a joint optimization over the calibration parameters between any number of cameras and unknown joint angles using a pose-loop error optimization approach, thereby avoiding the need for overlapping fields-of-view. We test our method in simulation and provide a calibration sensitivity analysis for different levels of camera intrinsic and joint angle noise. In addition, we provide a novel analysis of the degenerate parameters in the calibration when joint angle values are unknown, which avoids situations in which the calibration cannot be uniquely recovered. The calibration code will be made available at https://github.com/TRAILab/AC-DCC**

I. INTRODUCTION

The Simultaneous Localization and Mapping (SLAM) problem has been well studied, with significant advances enabling robots to be used in large-scale real-world applications both in indoor and outdoor settings. In [1], the authors state that the SLAM research community has entered an era focused on "*realizing robust performance, high-level understanding, resource awareness and task-driven perception.*" These requirements are particularly stringent for aerial vehicles with severe payload limitations. A common solution is to use a Static Camera Cluster (SCC) which consists of a group of cameras rigidly mounted with respect to each other [2]. While SCCs can be strategically arranged in order to achieve a panoramic view of the environment, the large amount of data available from many cameras imposes a bottleneck on the system and also causes the pose estimation quality to degrade when features in one or more cameras become occluded due to robot motion.

Dynamic Camera Clusters (DCCs) are multi-camera systems where one or more cameras are mounted to an actuated mechanism, such as the gimbal present on most commercially available drones, visible in Fig.1. DCCs allow the dynamic cameras to select the viewing direction, thereby gaining the advantage of omni-directional SCCs in terms of robustness, while maintaining a low camera count. Active vision [3] allows movable cameras such as DCCs to fixate and track previously detected landmarks over extended periods of time thereby improving the map and pose of the robot. However, the drawback of active DCC viewpoint selection, is that an accurate time-varying extrinsic transformation needs to be determined.

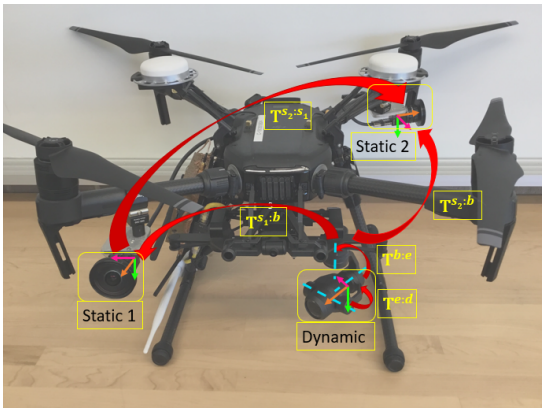

Fig. 1. A Dynamic Camera Cluster consisting of two static cameras and a dynamic camera attached to a 3-DOF gimbal on the 210 RTK DJI Matrice.

Previous DCC calibration methods [4], [5], [6] used a fiducial target where common points needed to be visible in both cameras for each measurement set, which was limiting in three aspects. First, a time-consuming process was undertaken in order to determine joint angles from which the target was visible in the dynamic camera and increased in complexity with the number of degrees of freedom of the mechanism. Second, while a calibration was achieved in a restricted range of motion, no guarantee of the calibrated parameters was provided outside the angle bounds used during the calibration process. When used in an active SLAM application, ensuring a precise calibration over the entire configuration space is important in order to limit inaccurate transformations of measurements from corrupting the optimization. Third, calibration involving more than two cameras with different overlapping FOV would require the tedious process of calibration to be performed multiple times.

Therefore, we present a method that is able to calibrate the time-varying extrinsic transformation between any number of cameras and achieves measurement excitation over the entire configuration space of the mechanism resulting in a more accurate calibration. To achieve this, we rely on the availability of the metric sensor poses which can be obtained

*Jason Rebello is a Ph.D. Candidate in the TRAILab at the University of Toronto Institute for Aerospace Studies (UTIAS). jason.rebello@robotics.utias.utoronto.ca
§Angus Fung is a MASc Candidate in the MIE department at the University of Toronto. angus.fung@mail.utoronto.ca
†Steven L. Waslander is an Associate Professor at the University of Toronto Institute for Aerospace Studies (UTIAS). stevenw@utias.utoronto.ca

via any stereo-vision, RGBD or Lidar SLAM process. We optimize over all the calibration parameters and unknown joint angles using the poses from each measurement set to obtain the calibration. We minimize the *pose-loop error* instead of the *pixel re-projection* error thereby eliminating the need for overlapping FOV. We test our calibration method in simulation and show that our method is able to achieve a better calibration resulting in a 10X lower pixel re-projection error. We also provide a sensitivity analysis by considering different levels and sources of noise for camera intrinsics and gimbal joint angles that can affect the calibration thereby providing insight into Dynamic Camera Cluster design.

In addition to a more accurate calibration, we identify the parameters when joint angle values are not available, that cause the system to enter a degenerate state, where the calibration cannot be uniquely recovered. For systems that do not possess dynamics, determining the degenerate parameters is equivalent to identifying the columns of the measurement jacobian the cause it to be rank deficient. We identify two degenerate configurations that arise when the joint angles for the first and last joint are added to the parameters to be estimated. As a final contribution, in order to accelerate research in Visual SLAM using DCCs, we release our calibration code at https://github.com/TRAILab/AC-DCC

II. Related Work

Dynamic Camera Cluster calibration is closely related to other manipulator calibrations such as hand-in-eye [7], head-to-eye [8] and kinematic calibration [9]. Hand-Eye and Kinematic calibration approaches generally use only a *single* camera, where as the DCC calibration computes time-varying extrinsics of *multiple* cameras which are related through an actuated mechanism. While these calibration procedures tackle parts of the DCC calibration, they do not account for the entire transformation from static camera to dynamic camera which is needed to resolve measurements between cameras in Visual Odometry and SLAM.

The calibration of a DCC was first performed in [4], where the forward kinematics of the actuated mechanism were parameterized using the well-known Denavit-Hartenberg convention and minimizing the re-projection error using overlapping viewpoints of a fiducial target. This was later extended in [5], to only select the most informative measurements that directly aid the calibration process. In [6], unknown encoder angles were added to the calibration procedure and later tested in a Visual Odometry application. All of the above methods relied solely on a fiducial target and limit the calibration to a pair of cameras. Our calibration procedure is capable of performing a single calibration optimization for any number of cameras and ensure an accurate calibration is obtained over the entire operating space of the mechanism.

There is an extensive body of work related to the calibration of Static Camera Clusters in natural environments. [10] develop a calibration procedure that is able to resolve an up-to-scale extrinsic transformation between cameras based on Visual SLAM. In order to resolve the scale of the calibration, [11] make use of odometry data that is easily available on ground vehicles while [12] incorporate a pair of stereo cameras on the vehicle. Our calibration procedure does not make any such assumption on the configuration of the cameras nor do we incorporate external sensors to resolve scale. Our work is most similar to [13] in the sense that we calibrate non-overlapping field of view cameras. However, [13] have the cameras rigidly mounted with respect to each other, whereas we account for the time-varying extrinsics in the DCC.

The analysis of degenerate configurations has been well studied with several works dealing with multi-camera [14], [15], [16], [17] and visual inertial[18] systems. In this work, we present an analysis of the calibration similar to [19], however, while the previous work analyzed the case with *known* joint angle values, our work tackles the degeneracy analysis when the joint angle values are *not available*, as is frequently the case with commercial drones.

III. Background and Notation

Frames and Notation: Let a point in 3D, expressed in co-ordinate frame $\mathcal{F}_x$, be denoted as $\mathbf{p}^x \in \mathbb{R}^3$, where $\mathbf{p} = [p_x \ p_y \ p_z]^T$ are the x, y and z components of the point, respectively. We define a rigid body transformation from frame $\mathcal{F}_a$ to $\mathcal{F}_b$ as $\mathbf{T}^{b:a} \in \mathbb{SE}(3)$, which is made up of a rotation $\mathbf{R}^{b:a} \in \mathbb{SO}(3)$ and translation $\mathbf{t}^{b:a} \in \mathbb{R}^3$ between the frames and is expressed in matrix form as,

$$\mathbf{T}^{b:a} = \begin{bmatrix} \mathbf{R}^{b:a} & \mathbf{t}^{b:a} \\ 0 & 1 \end{bmatrix} \tag{1}$$

Projection Model and PnP Solution: We define a projection model $\Psi(\mathbf{p}^c) : \mathbb{R}^3 \mapsto \mathbb{P}^2$ that maps a point expressed in camera frame, to a pixel location on the 2D image plane. Given a set of known 3D points in a co-ordinate frame $\mathcal{F}_w$ and its corresponding pixel location on the image plane, we can resolve the true pose of the camera in the frame $\mathcal{F}_w$ via the Perspective-n-Point (PnP) solution [20].

Denavit-Hartenburg Parameterization: We make use of the Denavit-Hartenberg (DH) convention to parameterize the kinematic chain which incorporates 4 independent parameters $\omega_l = [\, d_l,\, a_l,\, \alpha_l]^T$ and θ_l, where $\theta_l, \alpha_l \in [0, 2\pi)$ and $d_l, a_l \in \mathbb{R}$. The time-varying factor in this transformation is caused due to the joint value parameter θ_l. With successive co-ordinate frames defined for each of the joints, the transformation from frame $\mathcal{F}_i$ to $\mathcal{F}_{i-1}$ can be computed as follows:

$$\mathbf{T}^{i-1:i}_{\omega_l,\theta_l} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

For a more in-depth explanation on the DH parameterization, the reader is referred to [21].

Degeneracy Analysis: While the optimization of transformation matrices can be performed over $\mathbb{SE}(3)$, we perform the optimization and degeneracy analysis by treating the rotation part of the transformation matrix as a manifold in

$\mathbb{SO}(3)$, but using the translation component as a vector space in $\mathbb{R}^3$, similar to the analysis in [19]. This enables the use of the derivatives in [22] which leads to easier analysis of the degeneracies and rely on identities Eq.(3.17), Eq.(3.18) and Eq.(3.24) in [19], omitted due to space considerations. We let I be a 3x3 identity matrix and $[\mathrm{A}]_i$ represent the i^{th} column of the matrix A. We denote $[\cdot]^\wedge$ as the transformation of a vector to a skew symmetric matrix, while the $[\cdot]^\vee$ as the inverse of the corresponding operation. We make use of the following two identities for any rotation matrix $\mathbf{R}$ and any vector $\mathbf{v}$[22]:

$$[\mathbf{Rv}]^\wedge = \mathbf{R}[\mathbf{v}]^\wedge \mathbf{R}^T \quad (3) \qquad \mathbf{R}[\mathbf{v}]^\wedge = [\mathbf{Rv}]^\wedge \mathbf{R} \quad (4)$$

IV. PROBLEM FORMULATION

In Sec. IV-A, we briefly summarize the calibration procedure using the pixel re-projection error formulation as described in [6] with one static camera, s, and one dynamic camera, d. We then describe the pose-loop error formulation to calibrate the DCC for a dynamic camera and any number of static cameras in Sec. IV-B.

A. DCC Re-projection Error Calibration Formulation

The aim of the calibration process is to determine the time-varying rigid body transformation, $\mathbf{T}^{s:d}_{\Theta,\lambda}$, from the dynamic camera frame, $\mathcal{F}_d$, to the static camera frame, $\mathcal{F}_s$, where Θ and λ are the set of *estimated* parameters used to build the rigid body transform. Here λ is used to refer to the set of time-varying *unknown* joint angles and Θ refer to the set of static parameters constant for all measurement sets. For the calibration, the transformation between cameras has the form $\mathbf{T}^{s:d}_{\Theta,\lambda} = \mathbf{T}^{s:b}_{\tau_s}\mathbf{T}^{b:e}_{\omega,\lambda}\mathbf{T}^{e:d}_{\tau_d}$, where $\mathbf{T}^{s:b}_{\tau_s}$ and $\mathbf{T}^{e:d}_{\tau_d}$ are rigid 6-DOF transformations from the mechanism base frame to the static camera and the dynamic camera to the end effector respectively. $\mathbf{T}^{b:e}_{\omega,\lambda}$ defines a series of transformations from the end effector frame to the base of the mechanism parameterized by each link's DH parameters ω and λ as defined in Sec. III.

Since both cameras observe the same fiducial target with co-ordinate frame $\mathcal{F}_t$, the pose of each camera with respect to the target can be determined using the PnP transformation. Using this pose, points in the target frame can be expressed in each camera frame. Common points seen in both cameras can then be transformed from one camera, through the kinematic chain and projected onto the other camera enabling the calculation of the re-projection error as shown below.

$$e^s_j(\Theta,\lambda_i) = z^s_j - \Psi^s(\mathbf{T}^{s:d}_{\Theta,\lambda_i}\mathbf{p}^d_j) \quad (5)$$

where z^s_j is the pixel measurement of point j observed in the static camera and $\mathbf{p}^d_j$ is the 3D position of point j observed from the dynamic camera. λ_i refers to the set of joint angles for the current measurements set i. The total squared re-projection error as a function of the estimation parameters over all of the collected measurement sets, Z, is defined as

$$\Lambda(\Theta,\lambda) = \sum_{i=1}^{Z}\sum_{j=1}^{|P^d_i|} e^s_j(\Theta,\lambda_i)^T e^s_j(\Theta,\lambda_i) + e^d_j(\Theta,\lambda_i)^T e^d_j(\Theta,\lambda_i) \quad (6)$$

where $e^d_j(\Theta,\lambda_i)$ is the error calculated by projecting the same 3D point j expressed in the static camera, transformed through the chain and projected onto the dynamic camera image plane. Finally, an unconstrained optimization of Eq. (6) is performed in order to find the optimal calibration parameters, Θ^* and joint angles λ^*, which minimize the total re-projection error over the set of collected measurements,

$$\Theta^*, \lambda^* = \underset{\Theta,\lambda}{\operatorname{argmin}}\, \Lambda(\Theta,\lambda). \quad (7)$$

For a more in-depth review of the previous target based calibration approaches, the reader is referred to [4], [5], [6].

B. DCC Pose-loop Error Calibration Formulation

The quality of the calibration is heavily dependent on providing sufficient excitation to the joints inputs, as the parameter estimates are highly sensitive to the selected measurement sets. Since the re-projection error formulation required the same target point to be visible in both cameras, limitations were placed on the configurations for the gimbal from which to collect these measurements. Therefore, a pose-loop error as discussed below, was used in order to remove the dependence on overlapping FOVs and hence achieve a wider range of gimbal configurations. An added advantage of our method, is that the error operates on poses thereby abstracting out the method used to determine the camera poses in the reference frame.

Given the presence of *n* static cameras with co-ordinate frames $\mathcal{F}_{s_k}$, $k \in [1,..,n]$, and the location of 3D points in a target frame $\mathcal{F}_t$, the transformation from target frame to static camera $\mathcal{F}_{s_k}$ and dynamic camera $\mathcal{F}_d$ can be determined using the PnP solution. Let these transformations be denoted as $\mathbf{T}^{s_k:t}$ and $\mathbf{T}^{d:t}$ respectively. Therefore, given these transformations, it is possible to generate a measurement of the transformation from dynamic camera frame to each static camera frame $\tilde{\mathbf{T}}^{s_k:d}_i$ for measurement set i,

$$\tilde{\mathbf{T}}^{s_k:d}_i = \mathbf{T}^{s_k:t}(\mathbf{T}^{d:t})^{-1}, \quad \forall k \in [1,..,n] \quad (8)$$

Note that this transformation does not take into account any of the calibration parameters and is estimated purely from the visual measurements of the 3D points. Similar to Sec. IV-A, given some initial values of the calibration parameters along with estimates of the current joint angles from an Inertial Measurement Unit (IMU), we can determine the transformation from the dynamic camera to the static camera through the actuated mechanism $\mathbf{T}^{s_k:d}_{\Theta,\lambda_i}$.

Using these measurement sets, we can now determine the error $\varepsilon(\Theta,\lambda_i)$ between the measured transformation from dynamic to static cameras as expressed in Eq. (8) and the transformation from our initial guess of the calibration values. This difference in poses gives us the pose-loop error.

$$\varepsilon(\Theta,\lambda_i) = \mathbf{T}^e = \tilde{\mathbf{T}}^{s_k:d}_i \boxminus \mathbf{T}^{s_k:d}_{\Theta,\lambda_i}, \quad \forall k \in [1,..,n] \quad (9)$$

where the $\boxminus$ operator accounts for the difference of Transformation matrices which lie on a manifold, and returns a 6x1 vector.

$$\tilde{\mathbf{T}}^{s_k:d}_i \boxminus \mathbf{T}^{s_k:d}_{\Theta,\lambda_i} = [\log(\tilde{\mathbf{T}}^{s_k:d}_i(\mathbf{T}^{s_k:d}_{\Theta,\lambda_i})^{-1})]^\vee \quad (10)$$

While Eq. (9) provides an error term between the dynamic camera and each static camera, we can also impose an identity loop constraint between all static camera poses for a single measurement set.

$$\beta(\phi, i) = \mathbf{I} \boxminus \mathbf{T}_{\phi,i}^{s_1:s_1} \tag{11}$$

where $\mathbf{I}$ is a 4x4 Identity matrix and ϕ are parameters between the static cameras. By abuse of notation, $\mathbf{T}_{\phi,i}^{s_1:s_1}$ is a concatenation of transformations through the entire loop of static cameras and does not depend on the joint angles, $\mathbf{T}_{\phi,i}^{s_1:s_1} = (\mathbf{T}_{\phi,i}^{s_1:s_n})(\cdots)(\mathbf{T}_{\phi,i}^{s_2:s_1})$. The total squared pose-loop error for all measurement sets, W is then given by

$$\Lambda(\Theta, \phi, \lambda) = \sum_{i=1}^{W} \varepsilon(\Theta, \lambda_i)^T \varepsilon(\Theta, \lambda_i) + \beta(\phi, i)^T \beta(\phi, i) \tag{12}$$

We then optimize over Eq. (12) to determine the calibration parameters and unknown joint angles similarly to Eq.(7).

V. Degeneracy Analysis

In this section we present the degeneracy analysis related to the calibration of a 3-DOF Dynamic Camera Cluster with *unknown* joint angle values. It is important to note, that the degeneracy presented here is related to the parameterization and not the configuration of the actuated mechanism. The transformation chain from the dynamic camera to the static camera for the 3-DOF mechanism can be written as $\mathbf{T}^{S:D} = \mathbf{T}^{S:B}\mathbf{T}^{B:J2}\mathbf{T}^{J2:J3}\mathbf{T}^{J3:E}\mathbf{T}^{E:D}$. In particular we show that there are two degeneracies that arise between: (1) the transformation from base of the mechanism to the static camera ($\mathbf{T}^{S:B}$) and the encoder angle related to the first joint ($\mathbf{T}_{\theta_1}^{B:J2}$), and (2) the transformation from the dynamic camera to the end effector ($\mathbf{T}^{E:D}$) and the encoder angle related to the third joint ($\mathbf{T}_{\theta_3}^{J3:E}$).

The DH matrix in Eq. (2) is a concatenation of four separate matrices $\mathbf{T}_{\omega_l,\theta_l}^{i-1:i} = \mathbf{T}_{\theta_l}^{i-1:w}\mathbf{T}_{d_l}^{w:x}\mathbf{T}_{a_l}^{x:y}\mathbf{T}_{\alpha_l}^{y:i}$, where w, x, y are intermediate transformation frames [21]. We first present the jacobian of the DH matrix with respect to the unknown joint angle value (θ) parameter which causes a rotation around the z-axis. Using a similar analysis to Eq. (3.24) in [19], we see that adding a small perturbation to rotation around the θ parameter, $\psi_\theta = [0, 0, \delta_\theta, 0, 0, 0]$ results in the following

$$\mathbf{T}_\theta \boxplus \psi_\theta = \begin{bmatrix} \mathbf{R}_z(\delta_\theta) & 0_{3\text{x}1} \\ 0_{1\text{x}3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_z(\theta) & 0_{3\text{x}1} \\ 0_{1\text{x}3} & 1 \end{bmatrix} \tag{13}$$

$$= \begin{bmatrix} \mathbf{R}_z(\theta + \delta_\theta) & 0_{3\text{x}1} \\ 0_{1\text{x}3} & 1 \end{bmatrix} \tag{14}$$

Assuming the jacobian is first calcualted with respect to the rotation parameters and then translation, the previous equation shows that a perturbation δ_θ affects only the θ parameter and therefore affects only the third column of the composition jacobian. Using Eq.(3.17) and Eq.(3.18) from [19], we get

$$\frac{\partial \mathbf{T}_{\omega_l,\theta_l}^{i-1:i}}{\partial \mathbf{T}_{\theta_l}^{i-1:i}} = \begin{bmatrix} I & 0 \\ -[\mathbf{R}_{\theta_l}\mathbf{t}_{d_l,a_l,\alpha_l}]^\wedge & I \end{bmatrix}_3 = \begin{bmatrix} [I]_3 \\ -[\mathbf{t}_{\theta_l,\omega_l}^{i-1:i}]_3^\wedge \end{bmatrix} \tag{15}$$

where $\mathbf{R}_{\theta_l}\mathbf{t}_{d_l,a_l,\alpha_l} = [a_l \cos\theta_l \quad a_l \sin\theta_l \quad 0]^T$.

For the benefit of the reader we present the derivation for the jacobian of the first joint angle, with similar results for subsequent angles omitted for brevity.

$$\frac{\partial \mathbf{T}^e}{\partial \mathbf{T}_{\theta_1}^{B:J2}} = \frac{\partial \mathbf{T}^e}{\partial \mathbf{T}^{S:D}} \frac{\partial \mathbf{T}^{S:D}}{\partial \mathbf{T}^{S:J2}} \frac{\partial \mathbf{T}^{S:J2}}{\partial \mathbf{T}^{B:J2}} \frac{\partial \mathbf{T}^{B:J2}}{\partial \mathbf{T}_{\theta_1}^{B:J2}} \tag{16}$$

$$= \begin{bmatrix} \mathcal{J}_\boxminus^{S:D} & 0 \\ 0 & -I \end{bmatrix} \begin{bmatrix} I & 0 \\ -[\mathbf{R}^{S:J2}\mathbf{t}^{J2:D}]^\wedge & I \end{bmatrix} \begin{bmatrix} \mathbf{R}^{S:B} & 0 \\ 0 & \mathbf{R}^{S:B} \end{bmatrix} \begin{bmatrix} [I]_3 \\ -[\mathbf{t}^{B:J2}]_3^\wedge \end{bmatrix} \tag{17}$$

$$= \begin{bmatrix} [\mathcal{J}_\boxminus^{S:D}\mathbf{R}^{S:B}]_3 \\ [\mathbf{R}^{S:J2}\mathbf{t}^{J2:D}]^\wedge [\mathbf{R}^{S:B}]_3 + \mathbf{R}^{S:B}[\mathbf{t}^{B:J2}]_3^\wedge \end{bmatrix} \tag{18}$$

We now analyze the second row of Eq. (18). Using the fact that $\mathbf{R}^{S:J2} = \mathbf{R}^{S:B}\mathbf{R}^{B:J2}$ and the identity in Eq. (4), we get

$$\mathbf{R}^{S:B}[\mathbf{R}^{B:J2}\mathbf{t}^{J2:D}]_3^\wedge + \mathbf{R}^{S:B}[\mathbf{t}^{B:J2}]_3^\wedge \tag{19}$$

Using the vector space property of skew symmetric matrices and the transformation $\mathbf{T}^{B:D} = \mathbf{T}^{B:J2}\mathbf{T}^{J2:D}$, we arrive at,

$$\mathbf{R}^{S:B}[\mathbf{t}^{B:D}]_3^\wedge \tag{20}$$

We can therefore summarize the jacobians for the first and third joint angle values as

$$\frac{\partial \mathbf{T}^e}{\partial \mathbf{T}_{\theta_1}^{B:J2}} = \begin{bmatrix} [\mathcal{J}_\boxminus^{S:D}\mathbf{R}^{S:B}]_3 \\ \mathbf{R}^{S:B}[\mathbf{t}^{B:D}]_3^\wedge \end{bmatrix} \tag{21}$$

$$\frac{\partial \mathbf{T}^e}{\partial \mathbf{T}_{\theta_3}^{J3:E}} = \begin{bmatrix} [\mathcal{J}_\boxminus^{S:D}\mathbf{R}^{S:J3}]_3 \\ \mathbf{R}^{S:J3}[\mathbf{t}^{J3:D}]_3^\wedge \end{bmatrix} \tag{22}$$

We state the jacobians for base to static camera and dynamic camera to end effector below, and refer the reader to [19] for the derivation.

$$\frac{\partial \mathbf{T}^e}{\partial \mathbf{T}^{S:B}} = \begin{bmatrix} \mathcal{J}_\boxminus^{S:D} & 0 \\ [\mathbf{R}^{S:B}\mathbf{t}^{B:D}]^\wedge & -I \end{bmatrix} \tag{23}$$

$$\frac{\partial \mathbf{T}^e}{\partial \mathbf{T}^{E:D}} = \begin{bmatrix} \mathcal{J}_\boxminus^{S:D}\mathbf{R}^{S:E} & 0 \\ 0 & -\mathbf{R}^{S:E} \end{bmatrix} \tag{24}$$

In addition, [19] shows that for an actuated mechanism with *known* joint angle values, four degeneracies exist relating to the d_1, d_3, a_3 and α_3 parameters, where the number refers to the corresponding DH link parameters.

A. Degeneracy 1

This section presents the degeneracy that arises when the first joint angle value is added to the estimation parameters. More specifically, we show that when the base to static camera jacobian in Eq. (23) is multiplied with a constant $\begin{bmatrix} [\mathbf{R}^{S:B}]_3 \\ 0 \end{bmatrix}$ across all measurement sets, the result reduces to the jacobian of the first joint angle value as seen in Eq. (21).

$$\begin{bmatrix} \mathcal{J}_\boxminus^{S:D} & 0 \\ [\mathbf{R}^{S:B}\mathbf{t}^{B:D}]^\wedge & -I \end{bmatrix} \begin{bmatrix} [\mathbf{R}^{S:B}]_3 \\ 0 \end{bmatrix} \tag{25}$$

$$= \begin{bmatrix} [\mathcal{J}_{\boxminus}^{S:D}\mathbf{R}^{S:B}]_3 \\ [\mathbf{R}^{S:B}\mathbf{t}^{B:D}]^\wedge[\mathbf{R}^{S:B}]_3 \end{bmatrix} \quad (26)$$

Applying the identity in Eq. (3) to $[\mathbf{R}^{S:B}\mathbf{t}^{B:D}]^\wedge$ in the bottom half of Eq. (26), we get

$$\begin{bmatrix} \mathcal{J}_{\boxminus}^{S:D}[\mathbf{R}^{S:B}]_3 \\ \mathbf{R}^{S:B}[\mathbf{t}^{B:D}]_3^\wedge \end{bmatrix} = \frac{\partial \mathbf{T}^e}{\partial \mathbf{T}_{\theta_1}^{B:J2}} \quad (27)$$

This degeneracy arises due to the fact that the only constraint on assigning the co-ordinate frame to the base of the mechanism must have the z-axis along the joint axis of rotation. Therefore, the rotation around the z-axis is not constrained leading to an ambiguous absolute reference when trying to resolve the joint angle values. This degeneracy can be eliminated by removing one of the rotation parameters in $\mathbf{R}^{S:B}$ from the optimization.

B. Degeneracy 2

This section presents the degeneracy that results when we add the last joint angle as part of the parameters to be estimated. We show that by multiplying the jacobian of the dynamic camera to end effector in Eq. (24) with $\begin{bmatrix} [\mathbf{R}^{E:J3}]_3 \\ -[\mathbf{t}^{E:D}]^\wedge[\mathbf{R}^{E:J3}]_3 \end{bmatrix}$, which is constant for all measurement sets, a degeneracy in the jacobian can be identified.

$$\begin{bmatrix} \mathcal{J}_{\boxminus}^{S:D}\mathbf{R}^{S:E} & 0 \\ 0 & -\mathbf{R}^{S:E} \end{bmatrix} \begin{bmatrix} [\mathbf{R}^{E:J3}]_3 \\ -[\mathbf{t}^{E:D}]^\wedge[\mathbf{R}^{E:J3}]_3 \end{bmatrix} \quad (28)$$

$$= \begin{bmatrix} [\mathcal{J}_{\boxminus}^{S:D}\mathbf{R}^{S:J3}]_3 \\ \mathbf{R}^{S:E}[\mathbf{t}^{E:D}]^\wedge[\mathbf{R}^{E:J3}]_3 \end{bmatrix} \quad (29)$$

Inspecting Eq. (22) and Eq. (29), we see that the top half of the matrices match. Considering only the bottom term of Eq. (22) and the fact that $\mathbf{T}^{J3:D} = \mathbf{T}^{J3:E}\mathbf{T}^{E:D}$, we get

$$\mathbf{R}^{S:J3}[\mathbf{t}^{J3:D}]_3^\wedge = \mathbf{R}^{S:J3}[\mathbf{R}^{J3:E}\mathbf{t}^{E:D} + \mathbf{t}^{J3:E}]_3^\wedge \quad (30)$$

In [19], it was shown that the a parameter in the last DH link is degenerate. Therefore to simplify the jacobian, we can set a_3=0 and remove it from the estimated parameters. Given that skew-symmetric matrices lie in a vector space and the fact that $[\mathbf{t}^{J3:E}]_3^\wedge = 0_{3\text{x}1}$, we find that Eq. (30) gives

$$\mathbf{R}^{S:J3}[\mathbf{t}^{J3:D}]_3^\wedge = \mathbf{R}^{S:J3}[\mathbf{R}^{J3:E}\mathbf{t}^{E:D}]_3^\wedge \quad (31)$$

Using the identity in Eq. (3) on $[\mathbf{R}^{J3:E}\mathbf{t}^{E:D}]_3^\wedge$, we get

$$\mathbf{R}^{S:J3}[\mathbf{t}^{J3:D}]_3^\wedge = \mathbf{R}^{S:E}[\mathbf{t}^{E:D}]^\wedge[\mathbf{R}^{E:J3}]_3 \quad (32)$$

Substituting Eq. (32) in the bottom half of Eq. (22), leads to the matrix in Eq. (29), resulting in the degeneracy. This accounts for the fact that there is ambiguity in the relative rotation between the end effector and dynamic camera that is captured both by the 6-DOF transform $\mathbf{T}^{E:D}$ and last joint angle value. Eliminating any of $\mathbf{R}_y^{E:D}$,$\mathbf{R}_z^{E:D}$,$\mathbf{t}_x^{E:D}$,$\mathbf{t}_y^{E:D}$ or $\mathbf{t}_z^{E:D}$ from the optimization can resolve this degeneracy.

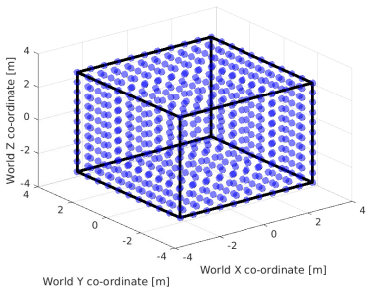

Fig. 2. Cube target environment for configuration test

VI. Experimental Validation

The calibration method in this paper was developed in order to perform a single joint optimization over any number of cameras while ensuring sufficient measurement excitation over the entire configuration space (c-space) of the gimbal. We therefore first perform a *Configuration Test* for a single static and dynamic camera by varying the joint angle limits from which measurements can be collected in order to test calibration quality for different levels of measurement excitation. We then perform a calibration for multiple static cameras and a dynamic camera attached to a 3-DOF mechanism in *simulation* with the setup similar to Fig. 1 and compare results using the target based and pose-loop methods. As a final experiment we test the calibration sensitivity for different levels of pixel and encoder noise. For all simulation experiments we incorporated real world pixel noise of 0.20 pixels and initialize the parameters to within $\pm$3 cm for translation and $\pm$20° for rotation unless specified.

A. Configuration Test

We perform simulated experiments to test the calibration quality achieved in two different environments, a 9x11 checkerboard target and 3D points generated on the surface of a 3x3x3 meter cube as shown in Fig. 2. The cube is used to represent a calibration area that incorporates natural features and could be achieved by any visual mapping procedure. We perform measurement collection in the presence of modest angle noise ($\pm$3°) over three different configuration spaces ($Target$, $Drone$, $Full\ Configuration(FC)$) and compare the calibration quality obtained for each space. The $Target$ configuration space requires the knowledge of joint limits from which the checkerboard could be seen and is set to [$\pm$20°] for all three axes. It should be noted that an increase in joint angle noise will reduce the configuration extent from which the target can be seen resulting in lower measurement excitation. The $Drone$ c-space allows the mechanism to collect measurements over a set of configurations that would typically be available on a drone and is set to [-120°:30°,$\pm$20°,$\pm$180°] for pitch, roll and yaw. For the FC configuration space, measurements are collected over the entire space without imposing any physical joint limits, allowing configurations throughout $\pm$180° for all axes. We performed the calibration procedure in Sec. IV-B and collected a total of 100 measurements randomly sampled in

TABLE I

CONFIGURATION TEST: MEAN AND STANDARD DEVIATION PIXEL ERROR FOR DIFFERENT CONFIGURATION SPACES.

	Target	Drone	FC
PE (mean)	4.65	0.91	0.57
PE (std)	6.9	1.3	0.65

the range of the joint angles for each configuration. For the test set, only the joint angles were optimized while keeping the calibrated parameters fixed. Given the true location of the static camera in the target frame and the optimized joint angles, the pose of the gimbal camera and therefore the pixel error can be calculated using the calibrations obtained from each of the methods resulting in the values shown in Table I. The choice of a pixel error evaluation is a more useful metric when dealing with measurements in feature space. As can be seen, the Target configuration has a 4X pixel re-projection error as compared to the Drone and FC configurations due to a lack of sufficient measurement excitation.

B. Simulated Multi-Camera Calibration

In this section we perform the DCC calibration for 2 static cameras and 1 dynamic camera similar to Fig.1, however this can be extended to any number of cameras. We assume the configuration where the static cameras do not have an overlapping FOV and hence need to perform the target based calibration for each pair of dynamic and static cameras. The target was placed in front of each pair and joint angle values for the dynamic camera corresponding to each static camera were noted. For the second static camera (SC2), the angle bounds were determined to be [-110°:-70°,±20°,±20°] while keeping the same angle bounds for the first static camera (SC1) as in Sec. VI-A. A total of 100 measurements were collected for each pair of cameras. For the calibration in the second environment, the entire drone was allowed to move in the cube and 100 measurements for the calibration and test set, consisting of each cameras pose in the map were collected over the entire Drone and FC c-space of the mechanism. The mean and standard deviation pixel error for the dynamic (PE-D) and second static camera (PE-S2) were noted in Table II. The target approach has more than a 10X higher pixel error despite using twice as many measurements as compared to the other approaches. This is due to the fact that the target method performs the calibration with each static camera separately and hence unable to determine a unique absolute joint angle reference for the first joint.

C. Sensitivity Analysis

In this section we present a simulation study of the most critical parameters affecting the calibration quality thereby providing insight into the considerations needed for building a DCC. We perform our analysis using the pose-loop error method and analyze the results over the Drone and FC c-space. We simulate 2 levels of pixel noise consisting of 0.3 and 1.2 pixel re-projection error that are typically obtained

TABLE II

SIMULATED MULTI-CAMERA TEST: MEAN AND STANDARD DEVIATION PIXEL ERROR FOR DIFFERENT CONFIGURATION SPACES.

	# Meas.	PE-D (mean)	PE-D (SD)	PE-S2 (mean)	PE-S2 (SD)
Target	140	21.80	21.66	6.89	0.27
Drone	70	1.97	2.14	0.16	0.01
FC	70	0.52	0.47	0.15	0.01

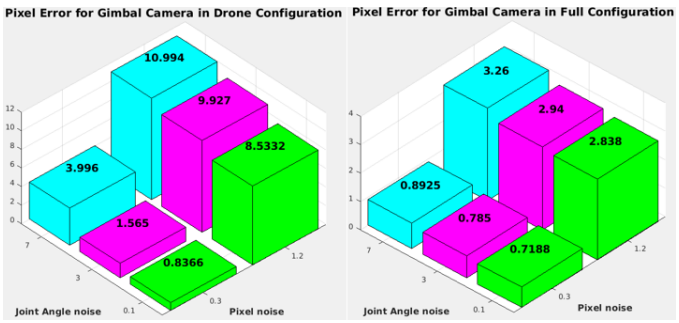

Fig. 3. Sensitivity analysis test comparing reprojection error for different levels of pixel and joint angle noise.

from a high quality and poor quality lens respectively. In order to account for a wide variety of gimbal mechanisms we simulate 3 levels of joint angle noise varying uniformly in the range of ±0.1°,±3° and ±7°. We collected a calibration and evaluation set of 70 measurements for all 3 cameras with joint angle values in the range specified in Sec.VI-A and noted the pixel re-projection error for the dynamic and second static camera as in Sec.VI-B. The resulting pixel error for the gimbal camera for the two configuration spaces are shown in Fig.3. As can be seen, we see a larger change in pixel re-projection error when moving from a low quality lens to a high quality lens for the same level of encoder noise. This is due to the fact that while open access to encoder values provides initial estimates to resolving the true pose of the dynamic camera, a higher measurement sensitivity for the calibration is achieved when using higher quality lenses thereby making it more crucial when designing DCCs.

VII. CONCLUSIONS

This paper demonstrated a method that is capable of calibrating a Dynamic Camera Cluster (DCC) consisting of any number of cameras while achieving sufficient measurement excitation guaranteeing a more accurate calibration compared to previous methods. Our proposed approach was tested in simulation consisting of a 3-DOF gimbal and multiple static cameras. In addition, we showed that incorporating the first and last joint angles into the estimation results in a degeneracy from which a unique calibration cannot be recovered. We provide a sensitivity analysis and make our code available to the research community in order to further development in DCC SLAM. Future work will consist of performing the calibration on real hardware as shown in Fig. 1 and incorporating the DCC in an Active Visual SLAM application and compared against Passive SLAM approaches.

REFERENCES

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. Leonard, "Past, Present, and Future of Simultaneous Localization And Mapping: Towards the Robust-Perception Age," *IEEE Transactions on Robotics (T-RO)*, vol. 32, no. 6, pp. 1309–1332, 2016.

[2] A. Tribou, Michael J. and. Harmat, D. Wang, I. Sharf, and S. L. Waslander, "Tracking and mapping with non-overlapping fields of view," *International Journal of Robotics Research*, vol. 34, no. 12, pp. 1480–1500, December 2015.

[3] R. Bajcsy, "Active perception," in *Proceedings of IEEE*, vol. 76, no. 8, 1988.

[4] A. Das and S. L. Waslander, "Calibration of a dynamic camera cluster for multi-camera visual SLAM," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, South Korea, October 2016.

[5] J. Rebello, A. Das, and S. L. Waslander, "Autonomous active calibration of a dynamic camera cluster using next-best-view," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[6] C. L. Choi, J. Rebello, L. Koppel, P. Ganti, A. Das, and S. L. Waslander, "Autonomous active calibration of a dynamic camera cluster using next-best-view," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[7] R. Horaud and F. Dornaika, "Hand-eye calibration," *The International Journal of Robotics Research*, vol. 14, no. 3, pp. 195–210, 1995.

[8] S. Kim, M. Jeong, J. Lee, J. Lee, K. Kim, B. You, and S. Oh, "Robot head-eye calibration using the minimum variance method," in *2010 IEEE International Conference on Robotics and Biomimetics, ROBIO 2010, Tianjin, China, December 14-18, 2010*, 2010, pp. 1446–1451.

[9] V. Pradeep, K. Konolige, and E. Berger, "Calibrating a multi-arm multi-sensor robot: A bundle adjustment approach," in *International Symposium on Experimental Robotics (ISER)*, 12/2010 2010.

[10] G. Carrera, A. Angeli, and A. J. Davison, "SLAM-based automatic extrinsic calibration of a multi-camera rig," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[11] L. Heng, B. Li, and M. Pollefeys, "Camodocal: Automatic intrinsic and extrinsic calibration of a rig with multiple generic cameras and odometry," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013, pp. 1793–1800.

[12] L. Heng, G. H. Lee, and M. Pollefeys, "Self-calibration and visual slam with a multi-camera system on a micro aerial vehicle," in *Proceedings of Robotics: Science and Systems*, Berkeley, USA, July 2014.

[13] A. Harmat, I. Sharf, and M. Trentini, "Parallel tracking and mapping with multiple cameras on an unmanned aerial vehicle," in *Proceedings of the International Conference on Intelligent Robotics and Applications (ICIRA)*, vol. 1, 2012, pp. 421–432.

[14] M. J. Tribou, D. W. Wang, and S. L. Waslander, "Degenerate motions in multicamera cluster slam with non-overlapping fields of view," *Image and Vision Computing*, vol. 50, pp. 27–41, 2016.

[15] J.-H. Kim, M. J. Chung, and B. T. Choi, "Recursive estimation of motion and a scene model with a two-camera system of divergent view," *Pattern Recognition*, vol. 43, pp. 2265–2280, 2010.

[16] B. Clipp, J.-H. Kim, J.-M. Frahm, M. Pollefeys, and R. Hartley, "Robust 6dof motion estimation for non-overlapping, multi-camera systems," in *Workshop on Applications of Computer Vision (WACV)*, Jan 2008, pp. 1–8.

[17] J. S. Kim and T. Kanade, "Degeneracy of the linear seventeen-point algorithm for generalized essential matrix," *Journal of Mathematical Imaging and Vision*, vol. 37, no. 1, pp. 40–48, May 2010.

[18] J. Kelly and G. S. Sukhatme, "Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration," *The International Journal of Robotics Research*, vol. 30, no. 1, pp. 56–79, 2011.

[19] A. Das, "Informed data selection for dynamic multi-camera clusters," in *PhD thesis, University of Waterloo, Mechanical and MEchatronics Engineering*, May 2018.

[20] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.

[21] R. S. Hartenberg and J. Denavit, *Kinematic Synthesis of Linkages*. McGraw-Hill, 1964.

[22] M. Bloesch, H. Sommer, T. Laidlow, M. Burri, G. Nuetzi, P. Fankhauser, D. Bellicoso, C. Gehring, S. Leutenegger, M. Hutter, and R. Siegwart, "A primer on the differential calculus of 3d orientations," in *https://arxiv.org/pdf/1606.05285.pdf*, October 2016.