Online eye-robot self-calibration

Arnaud Tanguy, Abderrahmane Kheddar, Andrew I. Comport

Online Eye-Robot Self-Calibration

Arnaud Tanguy[1,2], Abderrahmane Kheddar[3,2] and Andrew I. Comport[1]

***Abstract*— We present a new approach that extends the well known Eye-Hand calibration to the online whole-body calibration of the kinematic tree geometric parameters. Only on-board RGB-D sensor and joint encoders are required. Online calibration allows to estimate the state of the kinematic tree at any time, and thus account for inaccurate models, passive joints, mechanical wear, unexpected damages, *etc*. One major challenge in achieving such an online self-calibration procedure with the available sensors is that the observability of the calibrated parameters cannot always be guaranteed. In this work, we determine the effect of joint degrees of freedom on observability. From this, we propose a novel Eye-Robot calibration method that determines the geometric transformations between joints. Conditions on joint motion are further used to improve upon existing kinematic tree parameters when observability is incomplete. In practice a dense SLAM algorithm is used for online pose estimation and the results are demonstrated with an HRP-4 humanoid robot.**

I. Introduction

Robotic systems inherently require some sort of calibration procedure to determine parameters that are needed to perform state-estimation, planning and control. One of the most fundamental set of parameters is kinematics, which relates sensors to actuators through the geometric configuration of the robot. Whilst this model is most generally provided by the manufacturer thanks to precision machining, some parameters will change over time. This may be due to robot modifications and adaptations, passive mechanisms with no sensors, the addition of new sensors, or simply due to normal wear and tear. The accuracy and robustness of a robotic system is highly dependent on these calibration parameters and a solution to this problem is to develop life-long self-calibrating methods. In this paper a generic solution is proposed to continuously estimate these parameters using only encoder information and Dense Visual *VSLAM*, along with the observation of the contact configuration.

We exemplify our study with humanoid robots. They require accurate calibration to continuously interact with the environments. They are complex systems designed to be as versatile as humans in their interactions (*e.g.* locomotion in complex cluttered and uneven terrains) and manipulation (*e.g.* opening a valve, climbing a latter, drilling). They are also modelled with a large set of parameters, including the kinematic tree topology, link lengths, joint angles, sensor locations, *etc*. Reliable planning and control of their actions need these parameters to be determined with the accuracy required by the tasks to be achieved.

In the literature, apart from few recent papers, kinematic parameters are assumed to be known and unchanging, and mechanical links to be accurately calibrated by the manufacturer. Our contribution is to calibrate online the kinematic tree's geometric pose parameters using an on-board RGB-D sensor (for 6dof localisation and mapping) and joint encoders (optical). Floating-base 6D configuration (translation and rotation) *w.r.t.* to SLAM's environment map is also determined.

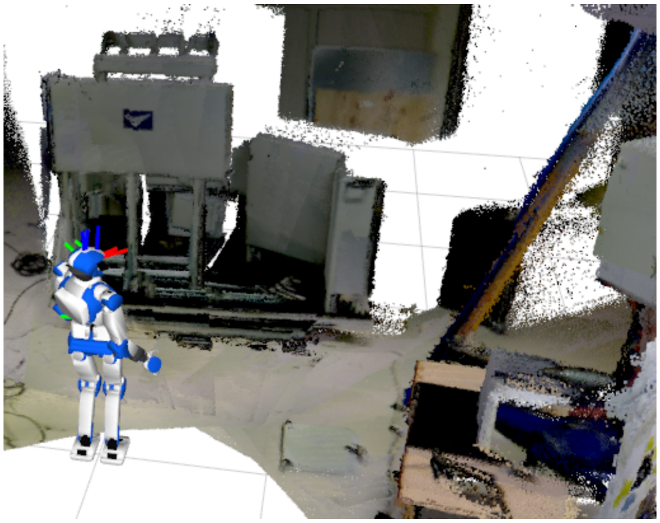

Fig. 1. Online Eye-Robot calibration of the kinematic tree geometric parameters for HRP-4's torso, head joints and camera (Hand-Eye). The floating base 6D (orienation and translation) is additionally calibrated *w.r.t.* the environment's point cloud.

II. Related work

There are several efficient methods for kinematic chain calibration offline. In this paper, a particular focus is made on methods suitable for online self-calibration, with minimal requirements for both the robot and the environment.

In [1] a method based on self-touch was proposed: using sensorized covers with thousands of tactile arrays, kinematic loops are formed by exploiting an existing imprecise forward kinematic model. Correspondences are established between predicted contact points and the sensed ones, which allows the refinement of kinematic parameters. While the idea is interesting, the sensory requirements are considerable.

Another common approach consists of forming a virtual closed loop between the camera and an end effector, by exploiting self-observation. In [2, 3], each end-effector's motion is observed by the robot's camera. As is the case with the sensory skin, this creates closed-loop constraints. Unfortunately this approach only corrects joint offsets and link lengths. These methods also suffer from two important drawbacks: observability issues, and the problem of robustly localizing the end-effectors. The later is often avoided by rigidly attaching calibration markers to the end-effectors during calibration [4, 5]. There are also additional uncertainties

*This work is supported by grant from the COMANOID H2020 project

[1] CNRS-University of Nice Sophia Antipolis, I3S, France

[2] CNRS-University of Montpellier, LIRMM, Interactive Digital Humans, France

[3] CNRS-AIST Joint Robotics Laboratory, UMI3218/RL, Japan

regarding the extrinsic pose of the calibration target *w.r.t.* the end effector.

Most recently, a generic online self-calibration algorithm was presented in [6]. It uses information theory to identify measurements that lead to improvements in calibration, and automatically detects and locks unobservable directions in parameter-space. This results in an online algorithm that listens to incoming sensor streams and builds a minimal set of data for estimating calibration parameters. The latter are continuously updated when they are observable; otherwise they are left to their initial guess. This is truly a remarkable work, however, it does not provide any guarantee that the parameters will eventually become observable. In calibrating a humanoid robot, one needs to make sure that the robot motions are suitable to guarantee full-observability of all kinematic parameters.

In order to perform Robot-Eye calibration it is necessary to estimate the pose of the camera. The most prominent method for determining the camera extrinsic calibration parameters is the Hand-Eye method. It commonly relies on calibration patterns, either fixed to the robot or placed within the environment, a choice mainly driven by lack of suitable markerless tracking systems. Recently, many state of the art dense visual simultaneous localization and mapping approaches [7, 8, 9, 10, 11] have been made available, and provide a robust pattern-free solution for tracking the camera pose (either RGB-D sensor, or a stereo pair), but also for reconstructing a 3D map of the environment, which will be of use in our approach. Any of these prominent SLAM solution is suitable for our proposed method. We opted for using the SLAM system of [9].

We propose a novel online method to calibrate and estimate each of the aforementioned parameter sets by making full use of dense visual *VSLAM*. Removing the need for calibration patterns and manual intervention we can achieve online corection of the calibration parameters at any time, including during normal robot operation. *Kinematic parameters* are determined by solving the proposed Robot-Eye calibration procedure which has been inspired by classic Hand-Eye calibration [12]. Similarly to [13] for hand-eye calibration, an analysis is performed on parameter observability to show that not all robot configurations are observable in a Robot-Eye setting. With full-observability, the kinematic parameters are obtained without need for any initial guess, which allows to calibrate fully unknown chains. When parameters are not fully observable, their calibration can still be improved along the observable directions, while keeping the remaining parameters to their inital guess. Our method is validated in both simulation, and with real experiments performed on and HRP-4 humanoid robot.

III. Robot-Eye Calibration of a Kinematic Chain

Our goal is to calibrate a robot's kinematic tree, provided that one has an estimate of the camera pose and joint state. In order to achieve this we show that the problem is related to that of hand-eye calibration. Yet standard hand-eye calibration methods are not sufficient. In particular, it assumes knowledge of the full 6dof motion of a robot link, which pressuposes a known kinematic model, which is exactly what we aim to determine in this work. First the classic approach is presented, then we develop our approach to Robot-Eye calibration that overcomes these limitations, while providing an online calibration procedure.

A. Hand-Eye calibration

Hand-Eye calibration was first described in [12], and has since become a standard tool for extrinsic calibration of vision sensors. It relates N relative sensor motions $\mathbf{A}_i \in \mathbb{SE}(3)$ with the corresponding motion $\mathbf{B}_i \in \mathbb{SE}(3)$ of a robot link to which it is rigidly attached to estimate the robot-sensor transformation $\mathbf{X} \in \mathbb{SE}(\mathbf{3})$. To simplify notations, the subscript i will often be ommitted. Each motion forms the following kinematic equality

$$\mathbf{A}_i\mathbf{X} = \mathbf{X}\mathbf{B}_i \tag{1}$$

- $\mathbf{A}_i$ is typically obtained by tracking the camera *w.r.t.* calibration patterns [12, 14].
- $\mathbf{B}_i$ is commonly obtained through forward kinematics, by assuming that the robot kinematic parameters are fully available.

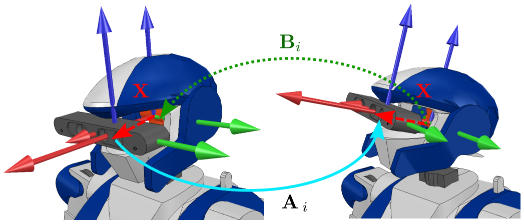

Fig. 2. Hand-Eye Calibration : the relative camera motion $\mathbf{A}_i$ and the relative robot motion $\mathbf{B}_i$ and the unknown extrinsic transformation $\mathbf{X}$ are forming a closed kinematic loop.

Several methods have been proposed to solve this equation. In [15], the rotation and translation are determined separately with standard non-linear least-square minimization. In [14], several formulations are proposed leading to multivariate polynomial optimization problems, that are globally solved using convex linear matrix inequality relaxations. A geometric analysis for the uniqueness of a solution is provided in [13], which shows that translation and rotation should not be decoupled. Following these recommendations from the literature, the solution to Equation 1 is determined by solving the following non-linear cost function (Equation 2), with the unknown transformation $\mathbf{X} \in \mathbb{SE}(3)$ parametrized using its lie algebra representation $\mathbf{x} \in \mathfrak{se}(3)$ such that $\mathbf{X} = e^{[\mathbf{x}]_\wedge}$, with $[.]_\wedge$ being the skew symmetric operator (see Equation 3).

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \sum_{i=1}^{N} ||(\mathbf{A_i} e^{[\mathbf{x}]_\wedge}{}^{-1} e^{[\mathbf{x}]_\wedge} \mathbf{B_i}|| \tag{2}$$

The Lie algebra representation allows simultaneously estimating both rotation and translation using a minimal number of 6 parameters (3 for rotation, 3 for translation), and robustness can be improved by using as many measurements as necessary to iteratively solve an over-determined system of equations.

In this paper, we exploit the tracking capabilities of dense visual SLAM systems to advantageously replace the measurements needed for the computation of $\mathbf{A}$. Using real-time SLAM localization offers two main advantages over standard methods: the dense method used is more accurate than conventional methods, and it can also be used to acquire data online. The Hand-Eye calibration problem can thus be formulated as an online calibration method, by solving Equation 1 from available robot and tracking data.

B. Calibration parameter observability

The observability of the pose $\mathbf{X}$, based on the observed movements $\mathbf{A}$ and $\mathbf{B}$, depends on the screw congruence theorem [13]. This theorem states that both $\mathbf{A}$ and $\mathbf{B}$ share the same screw transformation since they undergo the same movement. This is directly related to Mozzi-Chasles' theorem which states that the most general rigid body displacement can be produced by a translation along a line (called its screw axis) and a rotation about that line.

Here Chen's work is reformulated using a Lie Group representation. Consider the tangent spaces of the matrices $\mathbf{A}$ and $\mathbf{B}$ such that $[\boldsymbol{\alpha}]_\wedge = \mathrm{logm}(\mathbf{A}) \in \boldsymbol{se}(3)$ and $[\boldsymbol{\beta}]_\wedge = \mathrm{logm}[\mathbf{B}] \in \boldsymbol{se}(3)$. The tangent matrices $[\boldsymbol{\alpha}]_\wedge$ and $[\boldsymbol{\beta}]_\wedge$ are composed of both angular and linear components as follows:

$$[.]_\wedge = \begin{bmatrix} [\boldsymbol{\omega}]_\times & \boldsymbol{v} \\ 0 & 0 \end{bmatrix} \tag{3}$$

The axis of rotation is given by the line [16]:

$$\mathbf{l} = \frac{\boldsymbol{\omega} \times \boldsymbol{v}}{||\boldsymbol{\omega}||^2} + \lambda\boldsymbol{\omega} : \lambda \in \mathbb{R} \tag{4}$$

the rotation angle is:

$$\theta = ||\boldsymbol{\omega}|| \tag{5}$$

and the translation along the axis is:

$$\mathbf{t}^{||} = \frac{\boldsymbol{\omega}^\top \boldsymbol{v}}{||\boldsymbol{\omega}||} \tag{6}$$

Following the screw congruence theory [13], it can be shown that the rotation around the screw axis and the translation along the axis are invariant to coordinate frames such that $\mathbf{t}^{||}_\alpha = \mathbf{t}^{||}_\beta$ and $\theta_\alpha = \theta_\beta$. In the case of a 1dof rotational joint motion (as is the case for all the robots we consider), $\mathbf{t}^{||} = 0$ and $\theta = \widehat{\theta}$ is measured by the joint encoder. The relationship between the screw axis and its translation are, however, dependant on $\mathbf{X}$.

Subsequently it can be seen that the two invariant screw parameters do not provide any constraints for estimating the unknown calibration pose between the camera and any link of the robot. Only four parameters are constrained and the translation along and around the screw axis remains unobservable unless there is another orthogonal movement between the camera and the link (a minimum of 2dof are required to perform hand-eye calibration). Inevitably the hand-eye calibration of a 1dof joint (such as in a large number of robotic system) remains unobservable along those degrees of freedom. In the following sections, we will show how this knowledge about observability can be put into practice to calibration robots, even in cases where not all directions are observable.

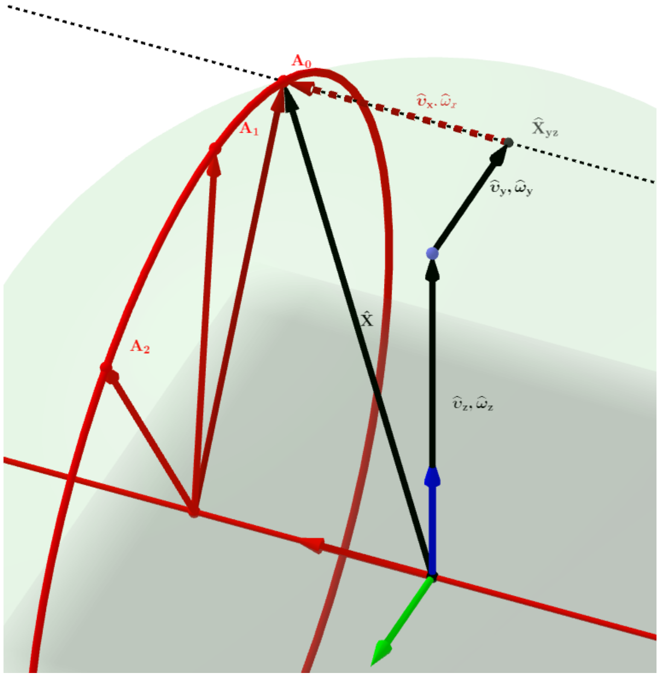
$\mathbf{A_0}$
$\mathbf{A_1}$
$\mathbf{A_2}$
$\widehat{v}_x, \widehat{\omega}_x$
$\widehat{\mathbf{X}}_{yz}$
$\widehat{v}_y, \widehat{\omega}_y$
$\widehat{\mathbf{X}}$
$\widehat{v}_z, \widehat{\omega}_z$

Fig. 3. Illustration of Hand-Eye calibration for a revolute joint. The joint is rotating around its x-axis, which generates sensor motion ($\mathbf{A}_0, \mathbf{A}_1, \mathbf{A}_2$). The Hand-Eye equations are not constrained along the joint's rotation axis, but are fully observable along the remaining directions y and z, for which a partial solution $\widehat{\mathbf{X}}_{yz}$ can be determined. The full solution lies along a line passing through the partial solution, with direction parallel to that of the joint's axis (*dashed black*). Translation and rotation along the joint's rotation axis are the only dof that cannot be determined (*dashed red vector*).

C. Eye-Joint calibration

As mentioned earlier, the most common use of Hand-Eye calibration has been in systems where both $\mathbf{A}$ and $\mathbf{B}$ have 6 degrees of freedom each. $\mathbf{B}$ is obtained by assuming the kinematic tree to be known and well-calibrated and its motion is computed by forward kinematics. As the goal is to extend the method to the calibration of an unknown kinematic chain, it cannot be determined this way. Given the sensors available (camera and encoders), the only parameters that may be fully determined are the camera motion, and the joint's intrinsic motion (*e.g.* rotation around its axis). We thus consider the special case of a camera rigidly linked to a single robot joint, and analyse solvability for common joint types. In this case, $\mathbf{B}$ represents the joint's intrinsic motion instead of the usual forward kinematic solution.

The solution $\mathbf{X}$ is only fully determined for *ball joints*, other joint types only admit a partial solution. This may at first seem very restrictive, until common mechanical designs for robot motions are considered. It is indeed common to find body parts that are moved by two revolute joints sharing the same rotation center (*e.g.* HRP-4's head, torso, hips, feet). In this case, the two joints considered together are equivalent to a single ball joint, and thus a full solution may be determined.

For revolute joints (*e.g.* HRP's knee joint), translation and rotation along the joint's rotation axis cannot be determined. As shown in Figure 3, the solution $\mathbf{X}$ may lie anywhere in both translation and rotation along the joint's rotation axis. The remaining parameters may however be fully observed. Consider a joint rotating around its x-axis.

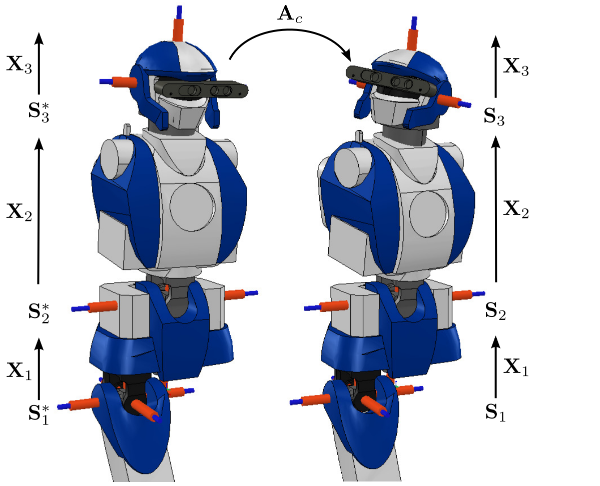

Fig. 4. Decomposed view of a kinematic chain motion between 2 postures. $\mathbf{A}_c$ is the measured camera motion through odometry, $\mathbf{S}_i$ represent each joint's intrinsic motion (the definition of $\mathbf{S}$ depends on the type of joint used). $\mathbf{X}_i$ is the rigid geometric transformation between the successive joints and is defined *w.r.t.* a reference robot posture. Notice that as is the case for Hand-Eye geometry, a closed-loop is formed, which allows for the estimation of the parameters, even through the camera motion is not necessarely defined *w.r.t.* the same reference frame as the robot motion.

Assuming a known initial guess $\mathbf{x}_x^* = [\upsilon_x^*, 0, 0, \omega_x^*, 0, 0]^T$ along the non-observable axis x, the remaining degrees of freedom (translation and rotation along y and z axes) can be determined by solving 2 for $\mathbf{x}_{yz} = [\upsilon_y, \upsilon_z, \omega_y, \omega_z]^T$ such that

$$\mathbf{x} = [\upsilon_x^*, \upsilon_y, \upsilon_z, \omega_x^*, \omega_y, \omega_z]^T$$

The solution $\hat{\mathbf{x}}$ for Equation 2 is then given as

$$\hat{\mathbf{x}} = [\upsilon_x^*, \hat{\upsilon}_y, \hat{\upsilon}_z, \omega_x^*, \hat{\omega}_y, \hat{\omega}_z]^T \quad (7)$$

That is, only the observable DoF are obtained, and the others are kept to their initial value, such as the known manufacturer's calibration if available. This makes it possible to improve calibration of the observable degrees of freedom online while keeping a known calibration (*e.g.* manufacturer's) for the others.

D. Online kinematic chain calibration

In the previous sections, we described the classic Head-Eye calibration method, and extended it to online Eye-Joint calibration. We now wish to extend it further to the calibration of a whole kinematic chain inducing a camera motion. Doing so first requires to define an initial reference state for the kinematic tree structure: joint states, and the geometric transformation between joints. Each joint is parametrized by a known intrinsic transformation $\mathbf{S_i} \in \mathbb{SE}(3)$, that depends on the type of joint used: rotation along the joint axis for a revolute joint, translation along axis for a prismatic joint, *etc.* The geometric transformations between joints are defined as the unknown parameters $\{\mathbf{X}_i \ / \ \mathbf{X}_i \in \mathbb{SE}(3)\}_{i=1..N}$ (see Figure 4), that we wish to minimize. The minimization vector is defined as their corresponding Lie-algebra twist representation $\mathbf{x} = \{\mathbf{x}_i \ / \ \mathbf{x}_i \in se(3)\}_{i=1..N}$. In order to solve for the unknown parameters $\mathbf{x}$ of this reference kinematic pose, the state of the robot in several postures needs to be acquired. We exploit the same argument as for Hand-Eye calibration: the geometric transformations between joints ($\mathbf{X}_i$) are rigid, thus constant. However, each joint intrinsic state varies. The standard Hand-Eye equation is extended by expressing the camera motion through all kinematic chain parameters as follows:

$$\mathbf{A}_c \mathbf{X}_N \mathbf{S}_N \ldots \mathbf{X}_1 \mathbf{S}_1 = \mathbf{X}_N \mathbf{S}_N^* \ldots \mathbf{X}_i \mathbf{S}_1^*, \quad (8)$$

where

- $\mathcal{S}^* = \{\mathbf{S}_N^* ... \mathbf{S}_1^*\}$ is the reference intrinsic joint state taken for the first measurement.
- $\mathcal{S} = \{\mathbf{S}_N ... \mathbf{S}_1\}$ is the current intrinsic joint state.
- $\mathbf{X}_i \mathbf{S}_i$ the instrinsic motion of joint i, followed by the geometric transformation of link i.
- $\mathbf{A}_c$ is the observed camera motion corresponding induced by the joints $\mathcal{S}$.

This formulation effectively decouples the geometric calibration parameters from the intrinsic joint state, and ensure that the solution is always provided at identity ($\mathcal{S} = \{\mathbf{I}_N \ldots \mathbf{I}_1\}$), which enables the calibration procedure to be started at any time while the robot is in motion.

Equation 8 is solved by non-linear Levenberg-Marquardt optimization stacking the errors obtained from each individual observed robot posture. However, as discussed in Section III-C, Eye-Joint calibration does not always admit a full solution, and thus the kinematic tree may only be partially determined.

IV. Results

Our method is evaluated on several kinematic chain configurations meant to illustrate the wide range of calibration results presented in this paper. First, using real data with HRP-4, we show that using the Eye-Joint method presented in Section III-C the extrinsic camera calibration can be obtained. We then illustrate the use of online Eye-Robot calibration in its full observability case by exploiting the ball-joint equivalence of HRP-4 torso and head joints. We show that the calibrated kinematic chain is consistent with our observability analysis in Section III-B. All real experiments are performed using the SLAM of [17], and the HRP-4 humanoid robot, controlled with a quadratic programming controller [18]. This result presentation is complemented by a video attachment showing the real-time online aspect of our proposed method.

A. Eye-Robot calibration of HRP-4

The HRP-4 humanoid robot has many degrees of freedom, an is fitted with very precise high-end optical encoders, and an Asus Xtion RGB-D sensor, making it a perfect candidate robot to test our proposed calibration techniques. All of HRP-4 joints are revolute, and many of its joints can be assimilated to fully-observable ball-joints, head rotates around yaw and pitch axes, so does the torso, and the leg waist joints. Other joints, such as the knee have only one degree of freedom.

1) Implementation: We release our open-source C++ implementation[1] of online eye-robot calibration. The implementation relies on the opensource C++ optimization framework

[1] `https://github.com/arntanguy/robcalib`

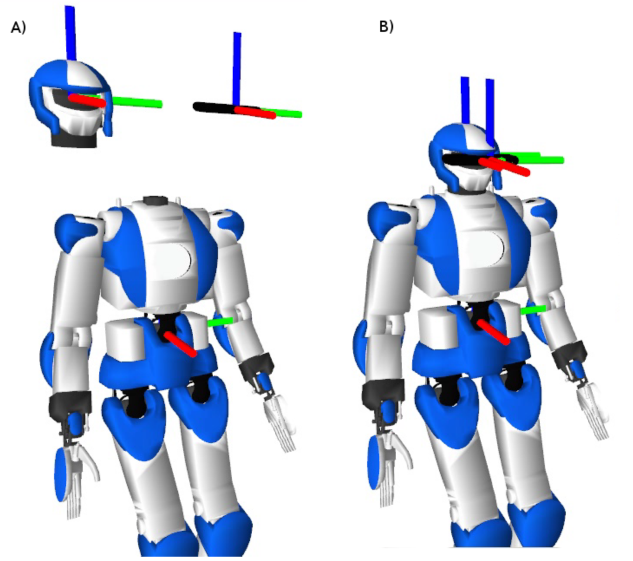

Fig. 5. Eye-Robot Calibration of HRP-4 waist-to-camera transformations. A/ The transformation between the torso joint and the head joint, and that between the head joint and the camera are initially unknown. B/ Results of the eye-joint calibration procedure obtained from a simulated robot motion, where both the torso and head joints are moved simultaneously. Camera pose along with encoder values are obtained from the simulated kinematic tree. Calibration results are continuously improved online as new data becomes available.

Roboptim[2] [19], which enables straightforward definition of optimization problems, and is compatible with a wide range of solvers. The cost function of Equation 8 is solved using the Levenberg-Marquardt solver of the Eigen library.

In practice, the equivalence between two successive revolute joints J_i and J_{i+1} sharing the same rotation center and a ball-joint is implemented by sharing the same optimization variable $\mathring{\mathbf{x}}_i$ for both joints, which removes 6 elements from the optimization vector. We are currently unable to automatically detect such joint configuration, and the responsibility for correctly defining the shared variables is left to the user.

2) Experiments: In a first experiment performed on the HRP-4 robot, we illustrate the equivalence between the classic Eye-Hand calibration, and the proposed Eye-Joint calibration to determine the camera-to-robot extrinsic calibration parameters. The initial camera-to-robot transformation is randomly chosen, and a motion is performed using both neck joints simultaneously, considered simultaneously as a ball-joint. As predicted by screw theory, the algorithm starts to converge with only two relative measurements, but greater accuracy is obtained over time. The complementary video further illustrate the online aspect of the calibration by estimating the location of the robot *w.r.t.* SLAM's map [20] using the live calibration results. As can be seen in the video and in Figure 1 the erroneous initial calibration results in the environment being misplaced *w.r.t.* the robot, while the end-result of the calibration correctly locates the robot.

In a second experiment, we show the Eye-Robot calibration method applied to the whole kinematic chain between the waist and the camera. In this configuration, the full geometric transformation between the torso and head joints, and head joint to the RGB-D sensor can be determined. Figure 5 shows the initial joint configuration along with the calibration results obtained after a few seconds of simulated robot motion. Both the torso and head joints are exited along their two degrees of freedom (rotation around the pitch and yaw axes) to guarantee full observability. Camera pose measurements are determined by forward kinematics of the ground-truth kinematic chain.

The same experiment is then performed on a real HRP-4 humanoid robot. Here, camera motion is obtained from dense visual SLAM [9], and joint angles from high-resolution optical encoders. As can be observed in Figure 1, aside from obtaining convincing calibration for the considered joints, the pose of the robot's floating base *w.r.t.* to SLAM's real-time map can also be determined. This is achieved by exploiting the known pose of the RGB-D sensor within SLAM's map, along with the kinematic chain being refined. Once convergence is reached, the robot attitude is determined. The complementary video further illustrate the online aspect of the calibration by estimating the location of the robot *w.r.t.* SLAM's map [20] using the live calibration results. As can be seen in the video and in Figure 1 the erroneous initial calibration results in the environment being misplaced *w.r.t.* the robot, while the end-result of the calibration correctly locates the robot.

B. Discussion

Calibration feasibility depends on the type of joints and their configuration. In particular, ball-joints parameters are fully observable. This is the case for many robots, in particular humanoids for which a common design is to have head, torso, and hips in a configuration akin to ball-joints. In this case, the proposed method is a direct generalization of the Hand-Eye calibration method to Eye-Joint calibration. For joints with only one degree of freedom, eye-joint calibration never admits a fully observable solution. In particular, for revolute joints, an ambiguity remains in both the translation and rotation along the joint's rotation axis. Such joint configuration is commonly found and we propose two main solutions for their calibration. The first focuses on calibrating the 4 observable DoF, which is achieved by locking the unobservable parameters (along the joint's axis) to an initial reference value, assumed to be available (Section III-C). Obtaining an accurate estimate of this initial value is however non-trivial, and one would most-likely fall back to using the manufacturer's calibration. For humanoid robots, the compromise of using the online calibration approach for the observable parameters, while using a known reference value for the others (manufacturer data, external calibration...) is best suited.

V. Conclusion and Future Work

Starting from Hand-Eye calibration results, we propose a novel method that performs online whole-body calibration of a kinematic chain using only joint encoders, and an RGB-D sensor. Contrary to Hand-Eye methods, no a-priori calibration of the kinematic chain is assumed available. It is necessary to replace the use of end-effector robot motion by that of the joints intrinsic motion, which leads to the

[2] http://roboptim.net

study of Eye-Joint geometry, where the robot motion is entirely determined by its intrinsic joint motion. Hand-Eye observability results are applied to this restricted problem, and we demonstrate that full observability of the geometric parameters is achieved for ball-joints, or equivalently two or more revolute joints sharing the same rotation center. We further consider solving Eye-Joint calibration for common joint types for which full observability cannot be achieved. In this case, screw theory is used to determine and calibrate the observable directions, while keeping an initial guess for the non-observable directions. Building upon these results, we then reformulate Hand-Eye calibration to a full kinematic chain calibration, denoted as Eye-Robot calibration, and successfully apply the method with both simulated and real experiments to the calibration of the upper-body of an HRP-4 humanoid robot. When full observability cannot be obtained, we demonstrate the ability to still improve the remaining degrees of freedom while keeping an initial calibration for the unobservable directions. Future work will focus on quantitative evaluation of the calibration results and of the influence of uncertainties in sensor measurements.

References

[1] A. Roncone, M. Hoffmann, U. Pattacini, and G. Metta, "Automatic kinematic chain calibration using artificial skin: Self-touch in the iCub humanoid robot," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 2305–2312, 2014.

[2] M. Hersch, E. Sauser, and A. Billard, "Online Learning of the Body Schema," *International Journal of Humanoid Robotics*, vol. 05, no. 02, pp. 161–181, 2008.

[3] R. Martinez-Cantin, M. Lopes, and L. Montesano, "Body schema acquisition through active learning," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 1860–1866, 2010.

[4] T. Kastner, T. Rofer, and T. Laue, "Automatic robot calibration for the NAO," *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, vol. 8992, pp. 233–244, 2015.

[5] V. Pradeep, K. Konolige, and E. Berger, "Calibrating a multi-arm multi-sensor robot: A bundle adjustment approach," in *Springer Tracts in Advanced Robotics*, vol. 79, 2014, pp. 211–225.

[6] J. Maye, H. Sommer, G. Agamennoni, R. Siegwart, and P. Furgale, "Online self-calibration for robotic systems," *The International Journal of Robotics Research*, vol. 35, no. 4, pp. 357–380, 2016.

[7] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "Dtam: Dense tracking and mapping in real-time," in *Proceedings of the 2011 International Conference on Computer Vision*. Washington, DC, USA: IEEE Computer Society, 2011, pp. 2320–2327.

[8] H. Zhou, K. Ni, Q. Zhou, and T. Zhang, "An SfM Algorithm With Good Convergence That Addresses Outliers for Realizing Mono-SLAM," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 2, pp. 515–523, 2016.

[9] M. Meilland and A. I. Comport, "On unifying keyframe and voxel-based dense visual SLAM at large scales," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Tokyo, Japan, 3-7 November 2013, pp. 3677–3683.

[10] T. Whelan, M. Kaess, M. Fallon, H. Johannsson, J. Leonard, and J. McDonald, "Kintinuous: Spatially extended KinectFusion," in *RSS Workshop on RGB-D: Advanced Reasoning with Depth Cameras*, Sydney, Australia, Jul 2012.

[11] J. Engel, J. Stueckler, and D. Cremers, "Large-scale direct slam with stereo cameras," in *International Conference on Intelligent Robots and Systems*, September 2015.

[12] R. Y. Tsai and R. K. Lenz, "A New Technique for Fully Autonomous and Efficient 3D Robotics Hand/Eye Calibration," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 3, pp. 345–358, 1989.

[13] H. Chen, "A screw motion approach to uniqueness analysis of head-eye geometry," *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 145–151, 1991.

[14] J. Heller, D. Henrion, and T. Pajdla, "Hand-eye and robot-world calibration by global polynomial optimization," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3157–3164, 2014.

[15] F. C. Park and B. J. Martin, "Robot Sensor Calibration: Solving AX = XB on the Euclidean Group," *IEEE Transactions on Robotics and Automation*, vol. 10, no. 5, pp. 717–721, 1994.

[16] R. M. Murray, S. S. Sastry, and L. Zexiang, *A Mathematical Introduction to Robotic Manipulation*, 1st ed. Boca Raton, FL, USA: CRC Press, Inc., 1994.

[17] M. Meilland, T. Drummond, and A. I. Comport, "A unified rolling shutter and motion blur model for 3D visual registration," *IEEE International Conference on Computer Vision*, pp. 2016–2023, 2013.

[18] J. Vaillant, A. Kheddar, H. Audren, F. Keith, S. Brossette, A. Escande, K. Bouyarmane, K. Kaneko, M. Morisawa, P. Gergondet, E. Yoshida, S. Kajita, and F. Kanehiro, "Multi-contact vertical ladder climbing with an HRP-2 humanoid," *Autonomous Robots*, vol. 40, no. 3, pp. 561–580, 2016.

[19] T. Moulard, F. Lamiraux, K. Bouyarmane, and E. Yoshida, "Roboptim: an optimization framework for robotics," in *Robomec*, 2013.

[20] A. Tanguy, P. Gergondet, A. Kheddar, and A. I. Comport, "Closed-Loop RGB-D SLAM Multi-Contact Control for Humanoid Robots," *Proceedings - IEEE/SICE International Symposium on System Integration*, 2016.