

Towards Body Schema Learning using Training Data Acquired by Continuous Self-touch

Qiang Li, Robert Haschke, and Helge Ritter

Abstract—To augment traditionally vision-based body schema learning with a sensory channel that provides more accurate positional information, we propose a tactile-servoing feedback controller that allows a robot to *continuously* acquire self-touch information while sliding a fingertip across its own body. In this manner one can quickly acquire a large amount of training data representing the body shape.

We compare three approaches to track the common contact point observed when one robot arm is touching the other in a bimanual setup: feed-forward control, solely relying on a CAD-based kinematics, performs worst; a controller that is only based on tactile feedback typically lacks behind; only the combination of both approaches yields satisfactory results.

As a first, preliminary application, we use the self-touch capability to calibrate the closed kinematic chain formed by both arms touching each other. The obtained homogeneous transform describing the relative mounting pose of both arms, improves end-effector position estimations by a magnitude compared to a traditional, vision-based approach.

I. INTRODUCTION

One precondition for a robot to generate reasonable motion behavior is that it can maintain an accurate internal representation – its body schema [5]. Exploiting this representation, the robot can realize long-term motion planning or short-term reactive feedback control just according to current perception. To this end body schema, as an interdisciplinary concept, is being concerned by more and more researchers in the robotics community. The kinematics model is an important aspect of the body schema, and learning or adapting the kinematics model (or parts of it) is an important prerequisite for successfully realizing Cartesian space motion.

Classically, external reference measurements required for hand-eye calibration are provided by machine vision, which requires to estimate intrinsic and extrinsic parameters of the camera, as well as to track and associate key points on the robot end-effector to determine their 3D position w.r.t. the camera [11], [6]. Typically a specific calibration object with a well recognizable marker pattern is mounted in a known reference pose w.r.t. to the end-effector to facilitate this task. In order to estimate the full-body shape of the robot, such a pattern should be extended over the whole robot surface, which is typically not feasible. Also, the spatial accuracy along the view axis of the camera is bounded due to stereo-vision limitations.

In order to overcome these limitations, alternative sensory channels were proposed for calibration. Mittendorf et al. employ an artificial skin composed of modular units

comprising (amongst other sensors) a 3-axis accelerometer measuring the orientation of a skin unit w.r.t. gravity. Performing specific motion patterns of individual joints, they can determine its kinematics parameters [10]; and deriving the local connectivity structure and exploiting the known geometry of the units, their relative position on the robot surface can be estimated too [9].

In order to provide absolute calibration, Roncone et al. proposed to collect reference measurements using a contact-based approach [12]. Instead of using visual features, the basic idea is to use the robot’s tactile or force feedback to physically close a kinematic chain towards various reference points in the world – by directly touching them. This approach becomes feasible with the advent of advanced tactile sensors providing both, high spatial and temporal resolution for fingertips [13], [4], [7] or even the whole robot surface [3], [2]. In industrial settings, e.g. in milling machines, this contact-based calibration paradigm provides superior calibration results for pose estimation of a work piece – typically employing a force-torque sensor in the end-effector.

Roncone et al. [12] studied self-touch on the iCub robot. Using its CAD kinematics model, they were able to achieve closure of the kinematics chain at discrete contact points on the forearm. However, unobservable errors (like joint backlash) limited the success rate of their approach. In order to pave the way for autonomous self-exploration of the whole robot body using continuous, exploratory self-touch motions, we propose to use a closed-loop servo controller that maintains a desired contact force while sliding a finger across the robot’s surface. In this manner, we could continuously sample a large amount of training data without risking damage of the robot.

Similar to [12], we do not attempt to learn the robot kinematics from scratch, but exploit the available CAD model as an initialization. Although the CAD model can be used to guide the exploring finger close towards the robot’s surface, it is not accurate enough to exactly establish contact – neither stopping before contact nor penetrating the surface. To overcome this gap, a force feedback controller is needed that acts on top of the kinematics-driven feed-forward controller.

For our preliminary application task we confine ourselves to the calibration of the relative base transformation of the two involved robot arms – assuming that the arm kinematics provided by the CAD model is accurate (which is typically true for industrial-grade robots). Tiny errors in the base transformation will sum up to centimeter level at the end-

The authors are with the Neuroinformatics Group / CITEC, Bielefeld University, Germany. {qli,rhaschke,helge}@techfak.uni-bielefeld.de

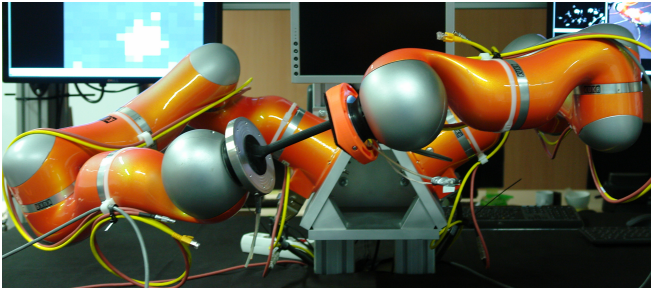


Fig. 1: Experimental setup for continuous self-touch: A rigid, not sensorized stick serves as a “finger” that explores the robot surface. Lacking a full tactile skin yet, the sensitive surface is limited to a flat tactile sensor array, mounted here on the end-effector of the other KUKA LWR arm.

effector, given a 1 m lever arm. The contribution of this paper is to propose a combined feed-forward and feedback servo control mechanism based on tactile feedback that guarantees a fast and safe sampling process to acquire closed-kinematic-chain training data. In order to realize fast sampling, we do not use *discrete* touch events as in [12], but we propose to employ our tactile servoing control framework [8] to allow for *continuous* exploration motions.

In that previous work we used a 8×8 cm tactile sensor array mounted at the end-effector of a KUKA LWR arm as a large “finger tip” exploring the unknown surface of an appropriately scaled-up object. In the present scenario, we are interested in closing the kinematic chain at the end-effectors of the robot arms. Hence we use a rigid, non-sensorized stick (serving as a passive finger on one arm) contacting the tactile sensor mounted as a “palm” on the other arm as shown in Fig. 1.

In order to collect reference points for calibration from different locations in the robot’s shared workspace, one can follow two strategies: On the one hand, the stick can slide across the sensor’s surface to contact different taxel locations (as we have shown in previous work). On the other hand, both arms can be moved simultaneously while maintaining contact (at the center of the sensor array). This obviously allows to cover a larger volume of the workspace as the sensing area is still restricted to the 8×8 cm sensor size.

In the latter case both end-effectors will move, and the basic idea of the proposed controller is to augment the previously developed tactile-servoing feedback controller [8] with a feed-forward controller that drives the slave arm according to the known motion of the master arm. Hence, the tactile-servoing controller only needs to correct tracking errors due to a wrong kinematics model. Using the proposed method, a robot can smoothly move both arms while maintaining the closed kinematic chain as required by [12].

The rest of the paper is arranged as follows: In next section, we introduce the proposed control scheme in detail. In section III, we introduce the proposed method to calibrate the relative mounting transform between both robot arms, which will be evaluated in section IV using a bimanual KUKA LWR robot setup. Finally, we give a conclusion and

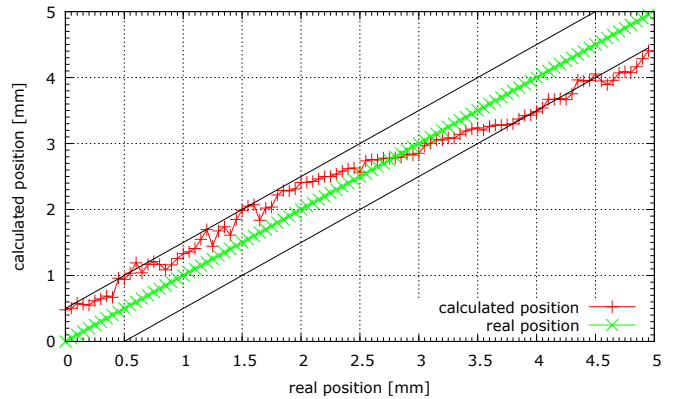


Fig. 2: Contact position (COP) when moving a 2mm-diameter probe tip from one to the other end of a single taxel. Due to the weighted averaging between different taxel locations (Eq. 1), we yield a spatial accuracy of ca. 0.5mm despite the much larger taxel size of 5mm.

an outlook on future work in section V.

II. CONTINUOUS SELF-TOUCH CONTROLLER

The proposed control approach draws on the availability of an artificial, tactile-sensitive skin providing spatially localized contact measurements. In our experiments we employ an array of 16×16 taxels with a spacing of 5mm in each direction. It is tuned towards high frame rates (up to 1.9 kHz), rendering a use for real-time robot control feasible [15]. The sensor exploits the piezo-resistive sensing principle, measuring changes in resistance of a conductive foam due to an applied force.

The tactile-servoing loop begins by computing the deviation of the current tactile feature vector \mathbf{f} from the desired one. In the present scenario, the tactile feature vector comprises the overall contact pressure p and the contact position \mathbf{c} which are determined from the sum of all taxel pressures p_{ij} within a contact region R and the pressure-weighted center of R :

$$p = \sum_{ij \in R} p_{ij} \quad \mathbf{c} = p^{-1} \sum_{ij \in R} p_{ij} \mathbf{c}_{ij}, \quad (1)$$

where \mathbf{c}_{ij} are the discrete coordinates of the taxels on the sensor array. Due to this averaging we can achieve a spatial resolution that is a magnitude better than the physical taxel spacing of 5mm, as can be seen in Fig. 2.

The feature error is fed into a P-controller, acting independently on all feature-error components. The resulting control variable \mathbf{u} is a (non-uniformly) scaled version of $\Delta \mathbf{f}$. Subsequently, the tactile feature error $\Delta \mathbf{f}$ is mapped onto a motion twist $\mathbf{V}_c^{\text{tact}} = [\mathbf{v}_c^{\text{tact}}, \boldsymbol{\omega}_c^{\text{tact}}]$ expressed in the contact frame, which will realize an error-reducing, relative motion between the involved end-effectors. To this end, the tactile servoing framework [8] employs the inverse sensor Jacobian \mathbf{J}_s^{-1} and a task-dependent projector matrix P selecting certain twist-components for control. In our application scenario, we only employ translation motions (neglecting rotations), such that

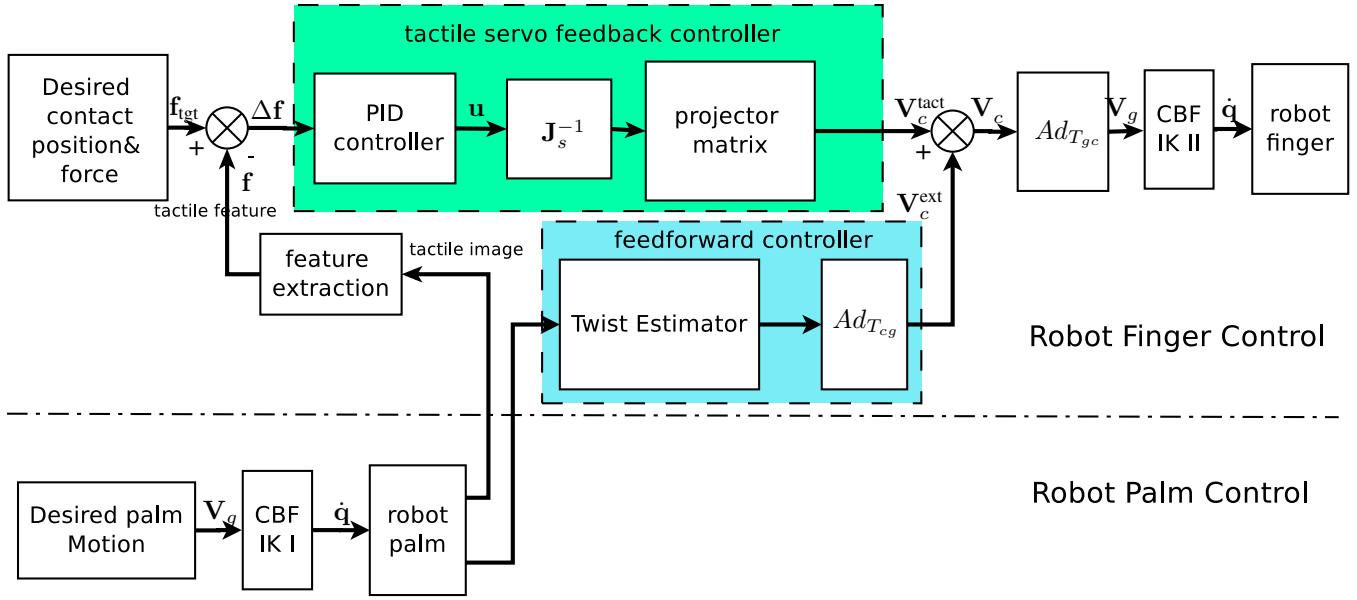


Fig. 3: Bimanual continuous self-touch control scheme.

$P \cdot \mathbf{J}_s^{-1}$ is a block-diagonal matrix of the form

$$P \cdot \mathbf{J}_s^{-1} = \begin{pmatrix} \mathbf{1}_{3 \times 3} & \\ & \mathbf{0}_{3 \times 3} \end{pmatrix}, \quad (2)$$

i.e. positional errors of the contact location will be corrected by a translation tangential to the sensor surface, while force errors will be corrected by a translation along the sensors normal direction. Summarizing, the feedback-part of the tactile servoing controller is determined by the following equation (also illustrated in the top box of Fig. 3):

$$\mathbf{V}_c^{\text{tact}} = P \cdot \mathbf{J}_s^{-1} \cdot K_P \cdot \Delta \mathbf{f}. \quad (3)$$

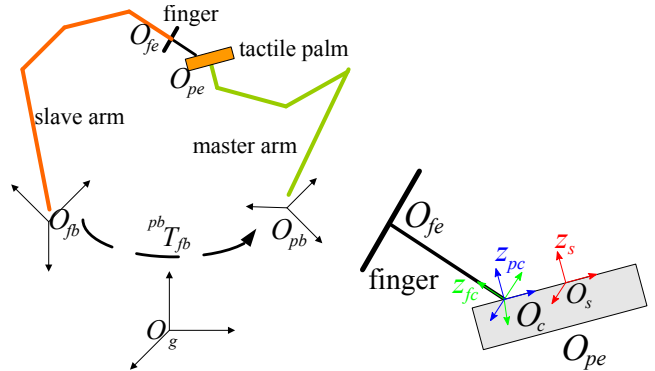
For more complex tactile-servoing methods we refer to [8].

Obviously, the tactile-servoing feedback controller cannot control the relative orientation of the stick to the tactile sensor – simply because the sensor can only measure normal forces, rather than the effective force direction. Hence, in order to maintain a perpendicular orientation of the stick to the sensor surface, we need to augment the tactile-feedback control with an orientation-based feedback control mechanism. Additionally, to better maintain contact position, we can forward-project the known palm motion (master) to the finger motion (slave). This motion estimation is performed by the lower controller box shown in Fig. 3, which is traditionally called “feed-forward controller” within our tactile servoing control framework as it provides a motion signal that is not determined from tactile feedback. However, in the present scenario it comprises a feedback loop on the Cartesian orientation of both end-effectors.

The linear and rotational motion components can be easily determined from the joint Jacobian \mathbf{J}_g and the orientation-correcting angular rotation vector $\hat{\omega}_{fp}$:

$$\mathbf{V}_g^{\text{palm}} = \mathbf{J}_g(\mathbf{q}) \cdot \dot{\mathbf{q}} \quad (4)$$

$$e^{\hat{\omega}_{fp}} = \mathbf{R}_{pc}^{-1} \cdot \mathbf{R}_{fc}, \quad (5)$$



(a) Closed kinematic chain. (b) Zoom into contact area.

Fig. 4: Definition of relevant coordinate frames.

where \mathbf{R}_{pc} and \mathbf{R}_{fc} denote the orientation matrices of the palm’s and finger’s contact frame w.r.t. the world frame (which are determined from the robot’s forward kinematics). The involved coordinate frames are illustrated in Fig. 4. The subscripts p and f are used to denote the palm and finger end-effectors respectively. The base and end-effector frames are denoted with the subscripts b and e , while the contact frame is denoted with subscript c . The origin of the contact frame is defined to be at the contact position. The origin of sensor frame O_s is defined to be in the center of the robot palm. The z -axes of the contact frames are always perpendicular to the sensor surface.

The individual twist vectors $\mathbf{V}_c^{\text{tact}}$, $\mathbf{V}_g^{\text{palm}}$, and $[0, \hat{\omega}_{fp}]$ are superimposed within a common coordinate frame (here the palm contact frame O_{pc}) and subsequently back-transformed into the global frame O_g using the adjoint matrix $Ad_{T_{gc}}$. Finally the powerful motion control framework of CBF [14] is employed to realize the computed contact frame motion with appropriate joint motions $\dot{\mathbf{q}}$, which completes the upper

part of the control scheme in Fig. 3.

The lower part deals with the deliberative motion of the palm mounted on the master arm. In our experimental scenario, the palm performs a hard-coded, open-loop Cartesian-space trajectory, which is actuated again using our control basis framework (CBF).

III. ROBOT JIG CALIBRATION

Kinematics errors will originate from several sources: First of all, from the relative transform between the two robot bases (denoted ${}^{pb}T_{fb}$ in Fig. 4a), second from a bad arm kinematics model, and third from a badly estimated tool transform. In the present paper we focus on the first aspect only as errors at the beginning of the kinematic chain have strongest influence on the positioning accuracy of the end-effector and we assume that manufacturing tolerances of the robot arm are negligible.

Hence, the goal is to improve the initial base transform (as estimated from visual calibration) by collecting sample contact location pairs $(\mathbf{p}_{fc}, \mathbf{p}_{pc})$ expressed w.r.t. to the slave or master arm base coordinate frames O_{fp} , and O_{pb} respectively. As the contact points should actually coincide for self-touch, the following equation should be fulfilled:

$$\bar{\mathbf{p}}_{pc} \equiv {}^{pb}T_{fb} \cdot \bar{\mathbf{p}}_{fc}, \quad (6)$$

where $\bar{\mathbf{p}}$ denotes homogeneous coordinates derived from Cartesian coordinates \mathbf{p} .

Having three or more non-colinear corresponding contact points, the relative transformation ${}^{pb}T_{fb}$ can be obtained as a solution to a least squares problem in a standard fashion [16].

IV. EXPERIMENT EVALUATION

We evaluate the proposed self-touch controller and the calibration method using the setup illustrated in Fig. 1 comprising two KUKA LWR arms with a “finger” stick and tactile “palm” mounted on their end-effectors. The arms are operated in joint-space compliance mode using a stiffness parameter of 1000 Nm/rad and a damping parameter of 0.7 Nm/s/rad.

We are using the in-house developed 16×16 tactile sensor array [15] that provides both, high spatial (0.5mm) and high temporal (1.9 kHz) resolution [8]. The robot control cycle is fixed to 250 Hz (using the KUKA FRI interface) and in sync with an appropriately reduced tactile feedback rate, which reduces high-frequency sensor noise and improves measurement accuracy.

While the tactile servoing framework in general provides PID-type control [8], here we only employ a P-controller (as indicated by Eq. 3) in order to compare different controllers in a fair way. The proportional coefficient is manually tuned to yield stable position and force tracking behavior.

A. Establishing contact

At beginning of an experiment, we manually position the finger opposite to the palm in a close, but not yet contacting pose, such that the finger stick points roughly

perpendicular to center of the tactile sensor. The very first task of the proposed self-touch controller is then to establish contact between the finger tip and the palm. To this end, a desired contact force is set as a target for the tactile-servoing controller. Because there is no initial contact, the measured contact force is zero, leading to a non-zero contact force error $\Delta \mathbf{f}$, such that a motion of the finger towards the palm is generated (along the normal direction of the palm surface as defined by Eq. 2). The approaching phase will continue until the contact force error becomes zero.

B. Actively sliding the finger across the sensor surface

Due to initial kinematics modeling errors (which we attempt to reduce), the contact point on the tactile sensor will typically deviate from the targeted center. Hence, the second task is typically to slide the finger towards the palm center while maintaining contact. To this end, the palm is still kept fixed and additionally to the force control primitive of the tactile servoing controller, the position control primitive is enabled [8].

To better illustrate the sliding capabilities, the accompanying video starts with a sequence, where the finger slides towards the edges of a square on the tactile sensor array. Particularly, the following taxel coordinates are approached in this sequence: center (8,8) \rightarrow lower right (3,3) \rightarrow upper right (3,12) \rightarrow lower left (12,3) \rightarrow upper left (12,12) \rightarrow center (8,8).

C. Continuous self touch tracking

To demonstrate the tracking capabilities when the master arm (the palm) moves as well, in the next video sequence, the palm actively moves in Cartesian space, while the finger is controlled to track the contact position at the center of the palm exploiting the proposed, composite feed-forward and feedback controller. For simplicity, the palm performs a translation motion along the x, y, z axes of the palm frame O_s , and rotates about its y and x axes. Along every motion direction, 10 training samples are collected for jig calibration.

D. Composite controller vs. feedback controller

Figures 5 and 6 compare the tracking performance of a plain tactile-servoing feedback controller with the proposed composite controller adding feed-forward projection of master arm motions. The palm is commanded a round trip motion along the palm’s x -axis with a peak velocity of 1 cm/s. The desired contact force is 1 N and the desired contact taxel coordinate is the center of the sensor. The figures show the force and position errors beginning from when the contact is detected. In the initial phase before the palm starts moving, there is no correction of contact position but only force control. Accordingly, the positional error remains fixed at its initial value – only slightly changing due to force correction motions.

At about 6 s resp. 5 s, the palm starts moving and the finger follows. At that point in time, positional errors are corrected using the feedback control mechanism and the error decreases in a similar fashion for both settings. At about 11 s,

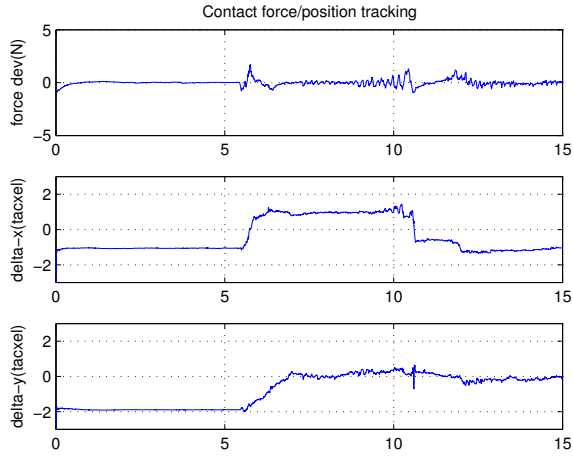


Fig. 5: Feedback controller for the translation experiment.

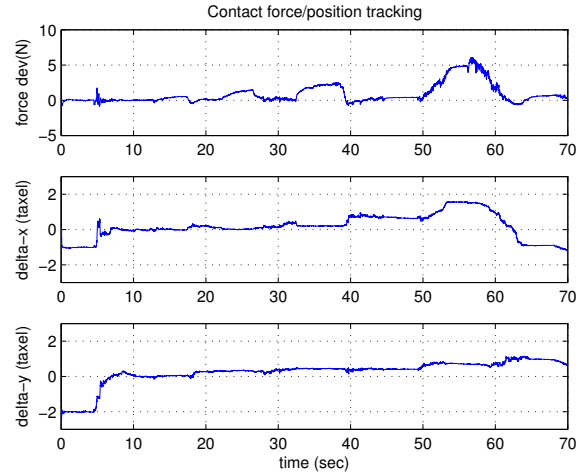


Fig. 7: Feed-forward controller for the rotation experiment.

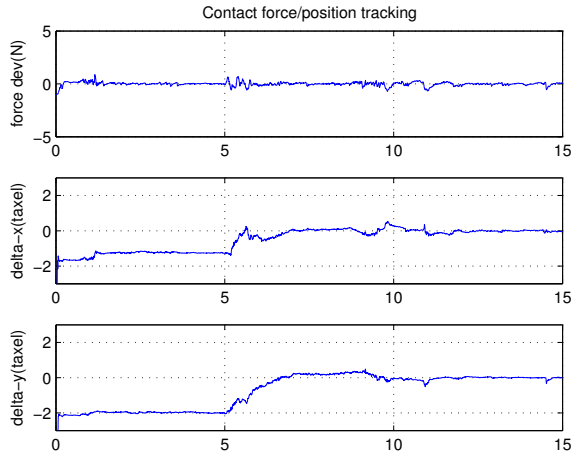


Fig. 6: Composite controller for the translation experiment

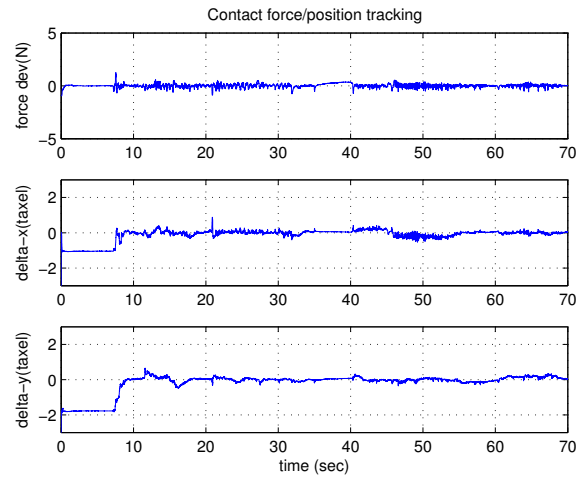


Fig. 8: Composite controller for the rotation experiment.

the palm starts moving back again, which can be nicely seen from the jumps in the error plots. Please notice, that the exact timing behavior is not identical between the experiments, as the palm motion was triggered manually.

However, as can be seen from Fig. 5, using only the feedback based controller, the finger significantly lags behind the palm, while with the additional forward projection, motion errors can be anticipated and directly corrected. Also, in purely feedback-based control, the positional error cannot be reduced to zero. This is due to the fact that we restricted the feedback controller to a P-term. Adding an integral component we could better compensate for systematic errors [1]. However, particular measures have to be taken to avoid i-term windup.

E. Composite controller vs. feed-forward controller

Figures 7 and 8 compare the proposed controller to a naive feed-forward controller, where the slave motion simply mimics the master's motion based on the internal kinematics model. In this comparison, the finger is first commanded to

slide towards the center of the tactile sensor (from 5 s to 13 s) in order to start both controllers at the same initial contact position. Subsequently, the palm is commanded the sequence of translation and rotation motions as described in sec. IV-C. Again, the desired contact force is 1 N, and the peak linear and angular velocities are 1 cm/s and 0.05 rad/s resp. From the plots it becomes clearly visible, that the proposed composite controller can compensate for kinematics errors and maintain the targeted position and force, while the feed-forward controller introduces significant errors.

F. Calibration Result

Using the training data sampled during the palm motion, we can determine the correct base transform between the two robot bases following the method described in sec. III. The deviation of the transform T_i obtained from tactile-based calibration compared to the one obtained using classical vision-based calibration (T_v) is given as:

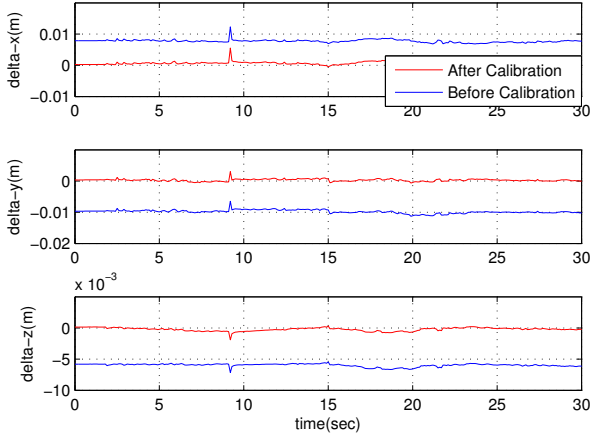


Fig. 9: Contact position error

$$T_t \cdot T_v^{-1} = \begin{pmatrix} 0.9999 & -0.0159 & 0.0009 & 0.0007 \\ 0.0159 & 0.9998 & -0.0077 & -0.0043 \\ -0.0008 & 0.0077 & 1.0000 & 0.0030 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

While this error-correcting matrix doesn't dramatically deviate from the identity matrix, it provides significant improvements to the kinematics model. Fig.9 shows the deviation of the contact position as observed from the kinematics model of the left and right robot arm – before and after tactile-based calibration. As can be seen from the plots, the errors reduced from about 1 cm along x , y axes and from about 5 mm along z -axis to sub-millimeter level.

V. CONCLUSIONS AND FUTURE WORK

Inspired by the contact-based kinematics calibration approach proposed in [12], we proposed a composite controller to allow for continuous self-touch exploration motions instead of only discrete contact events. Two continuous self-exploration strategies have been implemented: (i) sliding a finger on the robot's tactile-sensitive surface, and (ii) tracking a given contact point on a moving body segment. The proposed controller augments a previously developed tactile-servoing feedback controller with an anticipatory feed-forward controller, that can predict desired tracking motions from the motion of the moving body part. As the experimental comparison revealed, only the combination of both controllers was able to successfully track the contact location while maintaining a desired contact force.

Because the tactile sensor provides more accurate position references compared to standard vision (particularly along the camera's viewing direction), we could significantly improve the kinematics calibration (here only the base transform was considered), such that kinematics errors were reduced by a magnitude from centimeter-level to millimeter-level compared to the previously employed vision-based approach.

In order to extend the self-exploration capabilities towards whole-body surface exploration, we will in future employ our miniaturized tactile fingertip [7] to perform surface exploration motions in a similar fashion as reported in [8], thus collecting a detailed surface model of the whole robot body. However, this preliminary study has proven, that it is beneficial to prefer contact-based calibration to a vision-based approach.

ACKNOWLEDGMENT

This work was funded by SPP priority program "Autonomous Learning" (SPP 1527-1) funded by DFG. Authors would like to thank Dr. Matej Hoffmann and Alessandro Roncone for the fruitful discussions on tactile-based learning.

REFERENCES

- [1] Karl Johan Aström and Richard M Murray. *Feedback systems: an introduction for scientists and engineers*. Princeton university press, 2010.
- [2] Gereon Buescher, Martin Meier, Guillaume Walck, Robert Haschke, and Helge J. Ritter. Augmenting curved robot surfaces with soft tactile skin. In *Proc. IROS*. IEEE, 2015.
- [3] G. Cannata, M. Maggiali, G. Metta, and G. Sandini. An embedded artificial skin for humanoid robots. In *Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 434–438, 2008.
- [4] Jeremy A. Fishel. *Design and use of a biomimetic tactile microvibration sensor with human-like sensitivity and its application in texture discrimination using Bayesian exploration*. PhD thesis, University of Southern California, 2012.
- [5] Matej Hoffmann, Hugo Gravato Marques, A Hernandez Arieta, Hide-nobu Sumioka, Max Lungarella, and Rolf Pfeifer. Body schema in robotics: a review. *Autonomous Mental Development, IEEE Transactions on*, 2(4):304–324, 2010.
- [6] Hanqi Huang and Zvi S. Roth. On vision-based robot calibration. In *Southcon/94. Conference Record*, pages 104–109, March 1994.
- [7] Risto Kiva, Matthias Zenker, C Schrmann, Robert Haschke, and Helge J Ritter. A highly sensitive 3d-shaped tactile sensor. In *Proc. Advanced Intelligent Mechatronics (AIM)*, pages 1084–1089, 2013.
- [8] Q. Li, C. Schürmann, R. Haschke, and H. Ritter. A control framework for tactile servoing. In *Proc. RSS*, 2013.
- [9] P. Mittendorf and G. Cheng. 3d surface reconstruction for robotic body parts with artificial skins. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 4505–4510, Oct 2012.
- [10] P. Mittendorf, E. Dean, and G. Cheng. Automatic robot kinematic modeling with a modular artificial skin. In *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on*, pages 749–754, Nov 2014.
- [11] B Preising and TC Hisa. Robot performance measurement and calibration using a 3d computer vision system. *Robotica*, 13(04):327–337, 1995.
- [12] Alessandro Roncone, Matej Hoffmann, Ugo Pattacini, and Giorgio Metta. Automatic kinematic chain calibration using artificial skin: self-touch in the icub humanoid robot. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 2305–2312. IEEE, 2014.
- [13] Alexander Schmitz, Marco Maggiali, Lorenzo Natale, Bruno Bonino, and Giorgio Metta. A tactile sensor for the fingertips of the humanoid robot icub. In *Proc. IROS*, pages 2212–2217. IEEE, 2010.
- [14] M Schöpfer, F. Schmidt, M. Pardowitz, and H. Ritter. Open source real-time control software for the kuka light weight robot. In *8th World Congress on Intelligent Control and Automation (WCICA)*, 2010.
- [15] C. Schürmann, R. Kōiva, and R. Haschke. A modular high-speed tactile sensor for human manipulation research. In *World Haptics Conference*, 2011.
- [16] Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 13(4):376–380, 1991.