Low-cost Sensor Glove with Force Feedback for Learning from Demonstrations using Probabilistic Trajectory Representations

Elmar Rueckert*, Rudolf Lioutikov*, Roberto Calandra*, Marius Schmidt†, Philipp Beckerle† and Jan Peters*,‡

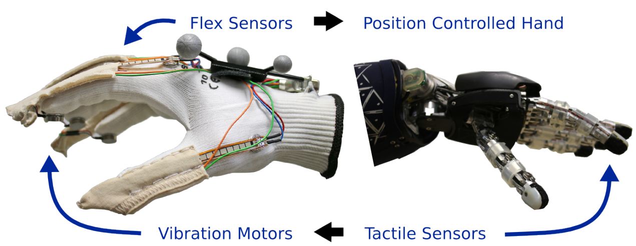

Fig. 1. A low-cost sensor glove is used to teleoperate a five-finger robot hand. The robot hand is equipped with tactile sensors (the *iCub* hand is shown in the picture). Tactile information provides force feedback to the teleoperator through activating vibration motors at the glove's fingertips.

***Abstract*— Sensor gloves are popular input devices for a large variety of applications including health monitoring, control of music instruments, learning sign language, dexterous computer interfaces, and teleoperating robot hands [1]. Many commercial products as well as low-cost open source projects have been developed.[1] We discuss here how low-cost (approx. 250 EUROs) sensor gloves with force feedback can be build, provide an open source software interface for Matlab and present first results in learning object manipulation skills through imitation learning on the humanoid robot *iCub*.**

I. INTRODUCTION

In robotics, sensor gloves are widely used for learning manipulation tasks from human demonstrations (see e.g., [2] for a recent overview). From the human operator perspective, active and passive approaches can be distinguished. In active approaches, the operator manipulates objects with its *own* hand. The robot learns a model through sensing the operator and the scene [3], [4], [5]. The demonstrations are platform independent and the learned model can adapt to changing environmental conditions. A mapping from human to robot kinematics and additional vision systems are needed.

Alternatively, in passive approaches the operator directly controls the robot hand through an instantaneous mapping from sensor glove measurements to control actions [6], [7]. This has the advantages that the human teleoperator can adapt for the limited capabilities of the robot (compared to humans) or to compensate for kinematic mapping errors. Moreover, joint and contact forces can be recored at the robot side and used to train inverse dynamics control methods. Additional force feedback at the sensor glove provides important information about contact forces during grasping. The drawback of passive approaches is that the demonstrations are platform specific and rapid learning methods are needed for teaching a large variety of robots new skills. We follow this line of research here.

Most models of demonstrations model single trajectories, either directly or indirectly (see e.g., [8] and [9]). A tracking controller with fixed or pre-tuned gains is used to follow the model prediction. However, for a safe operation of robots in everyday environments low gain control strategies are needed. Such compliant control approaches can be computed from the variance of the distribution over *multiple* trajectories. In probabilistic movement representations optimal control gains can be inferred that reproduce the demonstrations [10] or solve optimal control problems [11]. Here, we follow a simpler approach. As in probabilistic movement representations, a distribution over trajectories is learned using Bayesian linear regression with Gaussian features. Instead of inferring an optimal control law (which requires a forward model) we exploit a built-in low-level impedance controller to track the mean of the learned trajectory distribution. Variance dependent adaptive control gains approaches are part of future research.

II. METHODS

A. Sensor Glove Hardware

Our glove is based in the *animatronic robot hand project2* [2] [12], where we added force feedback. Finger motion data is acquired by 4.5 inch flex sensors measuring the bending of each finger, see Figure 1. The range of resistance is linearly mapped to the range of motions. To introduce force feedback, VPM2 coin vibration motors are placed at the fingertips. Those are controlled by mapping the tactile sensor readings of the robot hand proportionally to motor pulse width modulation (PWM) values. All sensors and motors are connected to an arduino board (mega 2560) that processes the data.

B. Glove Software Interfaces

The arduino board implements a serial communication interface (115200 Baud) and communicates with a computer

*Intelligent Autonomous Systems Lab, Technische Universität Darmstadt, Hochschulstr. 10, 64289 Darmstadt, Germany {rueckert, lioutikov, calandra }@ias.tu-darmstadt.de

†Institute for Mechatronic Systems in Mechanical Engineering, Technische Universität Darmstadt, Hochschulstr. 10, 64289 Darmstadt, Germany marius.schmidt@stud.tu-darmstadt.de, beckerle@ims.tu-darmstadt.de

‡Robot Learning Group, Max-Planck Institute for Intelligent Systems, Tuebingen, Germany mail@jan-peters.net

[1]The gloves project at http://theglovesproject.com/data-gloves-overview/ provides an overview.

[2]The mechanics and the electronics are detailed here: http://goo.gl/TqPYdK.

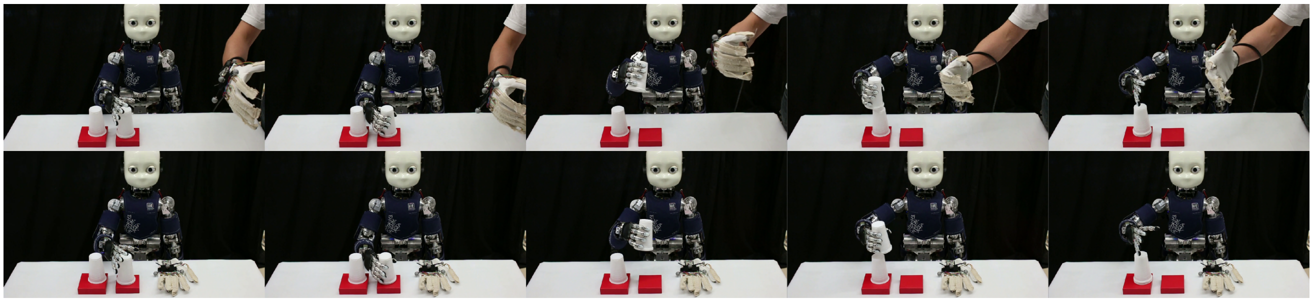

Fig. 2. Learning a cup stacking task. Top row: Demonstration through teleoperation. The robot arm is controlled using an approximation of the inverse kinematics following an end-effector trajectory provided by a motion capturing system. The sensor glove provides desired finger joint angles. Bottom row: Autonomous reproduction. An impedance controller is used to track the maximum a posteriori estimate of the trained probabilistic movement primitives.

through USB. In the current implementation, the communication protocol streams the flex sensor readings at a rate of 350Hz (with a resolution of two bytes per measurement). The sensor values can be accessed through a callback event raised by a Matlab *mex* function. Force feedback values can be send to the glove in form of a string command representing pulse width modulation (PWM) values $\in [0, 255]$.[3]

C. Learning from Demonstrations

We denote a single demonstration as sequence of T state vectors $\boldsymbol{\tau} = \boldsymbol{y}_{1:T}$. The state vector is modeled in a linear basis function model assuming zero mean Gaussian noise where $\boldsymbol{y}_t = \boldsymbol{\Phi}_t\, \boldsymbol{w} + \boldsymbol{\epsilon}_y$ with $\boldsymbol{\epsilon}_y \sim \mathcal{N}\left(\boldsymbol{\epsilon}_y \,|\, \boldsymbol{0}, \boldsymbol{\Sigma}_y\right)$. The matrix $\boldsymbol{\Phi}_t$ denotes the (extended) feature matrix using Gaussian basis functions [13].

To model a distribution over multiple trajectories we introduce the prior distribution $p(\boldsymbol{w}) = \mathcal{N}\left(\boldsymbol{w} | \boldsymbol{\mu}_w, \boldsymbol{\Sigma}_w\right)$. The mean $\boldsymbol{\mu}_w$ and the covariance matrix $\boldsymbol{\Sigma}_w$ can be learned from data by maximum likelihood using the Expectation Maximization (EM) algorithm [14]. A simpler solution that works well in practice is to compute first the most likely estimate of $\boldsymbol{w}^{[i]}$ for each trajectory $\boldsymbol{\tau}^{[i]}$ independently (where the index i denotes the i-th demonstration). Given a trajectory $\boldsymbol{\tau}^{[i]}$, the corresponding weight vectors $\boldsymbol{w}^{[i]}$ can be estimated by a straight forward least squares estimate $\boldsymbol{w}^{[i]} = \left(\boldsymbol{\Phi}_{1:T}^T \,\boldsymbol{\Phi}_{1:T} + \lambda\, \boldsymbol{I}\right)^{-1} \boldsymbol{\Phi}_{1:T}^T \;\; \boldsymbol{\tau}^{[i]}$. Subsequently, the mean and the covariance of $p(\boldsymbol{w})$ can be estimated by the sample mean and sample covariance of the $\boldsymbol{w}^{[i]}$'s.

After learning, the probability of trajectory $\boldsymbol{\tau}$ given the feature vector $\boldsymbol{w}$ reads $p(\boldsymbol{\tau} \,|\, \boldsymbol{w}) = \prod_{t=1}^{T} \mathcal{N}\left(\boldsymbol{y}_t | \boldsymbol{\Phi}_t\, \boldsymbol{w}, \boldsymbol{\Sigma}_y\right)$. For more details we refer to [10], [13]. Demo Matlab code was made open source.[4]

III. Cup Stacking Task on the iCub

The sensor glove was used to teach the humanoid robot *iCub* how to stack two plastic cups as depicted in Figure 2. Prior to the demonstrations, we recorded the minima and maxima of the five flex sensors when the teleoperator flexed and spread its fingers. These values were used to in a linear mapping from flex readings to the robot finger joint angles (the joint angle extrema of the robot fingers were manually determined). The arm joints were controlled through an inverse kinematics approximation given the Cartesian coordinate and orientation of the operator's wrist (measured in a motion capturing system). A built-in low-level impedance controller with manually tuned gains was used to track the desired joint angles during teleoperation or after training in the autonomous execution phase. The control rate was 200Hz and the movement duration was set to 15 seconds.

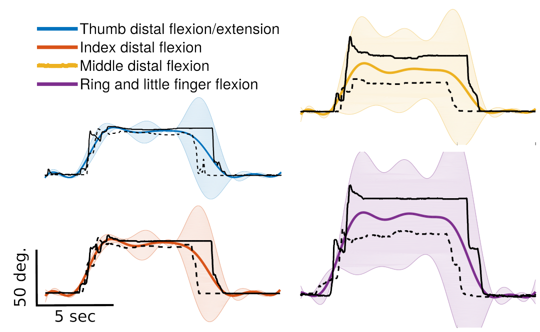

Fig. 3. Illustration of the joint encoder readings of four distal finger joints for two demonstrations (solid and dashed black lines). The little and the ring finger are coupled in the five finger hand. The model mean is denoted by the smooth colored lines. It is used in the autonomous reproduction phase in a tracking controller. The standard deviation is denoted by the shaded area.

Two demonstrations were used for training the probabilistic model. In total, a model of 13 joints was learned. For the distal finger joints the demonstrations and the learned model are shown in Figure 3. The model filters the demonstrations through averaging over the demonstrations. Note that the force feedback was helpful for grasping the cup without deforming it. This needs to be further evaluated. A successful autonomous execution tracking the mean of the learned distribution is shown in the bottom row in Figure 2.

IV. Conclusions

A low-cost sensor glove with force feedback was presented. We used it in first experiments for imitation learning in a cup stacking task in the humanoid robot *iCub*. We developed open source code for the sensor glove and the probabilistic model. Future hardware extensions include an

[3] Matlab code for the glove: `http://tinyurl.com/nf96nr7`, a Java interface: `http://tinyurl.com/o4hjumq`.

[4] Matlab code of the model: `http://tinyurl.com/qj4joxz`.

inertial measurement unit (to replace the motion capturing system) and additional flex sensors to measure thumb adduction/abduction and proximal joint motions. We also plan to exploit the learned variance in the probabilistic model for adaptive control gains.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement No. 600716 (CoDyCo). The authors would like to acknowledge Alexander Burkl, Chao Su, and Ji-Ung Lee for their assistance during the glove development.

REFERENCES

[1] David J Sturman and David Zeltzer. A survey of glove-based input. *Computer Graphics and Applications, IEEE*, 14(1):30–39, 1994.

[2] Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and autonomous systems*, 57(5):469–483, 2009.

[3] Chao-Ping Tung and Avinash C Kak. Automatic learning of assembly tasks using a dataglove system. In *Intelligent Robots and Systems 95.'Human Robot Interaction and Cooperative Robots', Proceedings. 1995 IEEE/RSJ International Conference on*, volume 1, pages 1–8. IEEE, 1995.

[4] Rüdiger Dillmann, Oliver Rogalla, Markus Ehrenmann, R Zollner, and Monica Bordegoni. Learning robot behaviour and skills based on human demonstration and advice: the machine learning paradigm. In *ROBOTICS RESEARCH-INTERNATIONAL SYMPOSIUM-*, volume 9, pages 229–238, 2000.

[5] Kathrin Gräve, Jörg Stückler, and Sven Behnke. Learning motion skills from expert demonstrations and own experience using gaussian process regression. In *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)*, pages 1–8. VDE, 2010.

[6] Max Fischer, Patrick van der Smagt, and Gerd Hirzinger. Learning techniques in a dataglove based telemanipulation system for the dlr hand. In *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, volume 2, pages 1603–1608. IEEE, 1998.

[7] Marco Henriques. Planning and control of anthropomorphic robotic hands using synergies. technical report.

[8] S. Schaal, J. Peters, J. Nakanishi, and A. Ijspeert. Learning movement primitives. In *International Symposium on Robotics Research (ISRR2003)*, 2004.

[9] J. Kober and J. Peters. Policy search for motor primitives in robotics. (1-2):171–203, 2011.

[10] A. Paraschos, C. Daniel, J. Peters, and G Neumann. Probabilistic movement primitives. In *Advances in Neural Information Processing Systems (NIPS), Cambridge, MA: MIT Press.*, 2013.

[11] E. Rueckert, G. Neumann, M. Toussaint, and W. Maass. Learned graphical models for probabilistic planning provide a new class of movement primitives. (97), 2013.

[12] Albert De Beir, Emilie Caspar, Florence Yernaux, Pedro Magalhaes Da Saldanha da Gama, Bram Vanderborght, and Axel Cleermans. Developing new frontiers in the rubber hand illusion: Design of an open source robotic hand to better understand prosthetics. In *Robot and Human Interactive Communication, 2014 RO-MAN: The 23rd IEEE International Symposium on*, pages 905–910. IEEE, 2014.

[13] E. Rueckert, J. Mundo, A. Paraschos, J. Peters, and G. Neumann. Extracting low-dimensional control variables for movement primitives. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2015.

[14] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.