Multisensory Assisted In-hand Manipulation of Objects with a Dexterous Hand

Timo Korthals[†*], Andrew Melnik[†*], Jürgen Leitner[‡], and Marc Hesse[†]

Deep Reinforcement Learning techniques demonstrate exciting results in robotic applications such as dexterous in-hand manipulation. One of the challenging factors which come into play when performing real-world tasks is a combination of different sensory modalities (vision, preconception, and haptic) which support each other for identification of the position and orientation of the manipulated object. This mutual support is important when some of the modalities are occluded or noisy. Furthermore, update rates of different sensory modalities may not match each other. While we assume that vision alone can determine the state perfectly, it suffers from slow update rates and it is susceptible to drop-out due to visual occlusion (e.g. palm over the object). On the other hand, haptic by means of touch and proprioceptive information is always present with a high update rate but suffers from ambiguous perception (e.g. the cube in Fig. 1 can take various possible orientations without a change in the haptic and proprioceptive perception). Therefore, we present an approach to infer the state of the object through a unified, synchronized, multisensory perception of position and orientation of a manipulated object.

Our approach builds upon the recent work of learning dexterous in-hand manipulation [1], where an agent with a model-free policy was able to learn complex in-hand manipulation tasks using proprioceptive and touch feedback plus visual information about the manipulated object. In this work, the agent can perform vision-based object reorientation on a physical Shadow-Hand in a simulated environment.

However, pose reconstruction of a manipulated object via vision is much slower than tactile or proprioceptive readings. Thus additional accuracy of the manipulated object's pose can be gained with proprioceptive and haptic information during visual drop-out, which motivates our current work.

I. APPROACH

Since we assume different temporal resolution of sensory inputs, but want to apply the standard approach by [1] to infer actions, we need to reconstruct the lacking vision based observation. The policy network g by [1] demands an observation $o = (o_\text{P}, o_\text{T}, o_\text{O})$ consisting out of the proprioceptive $o_\text{P} \in \mathbb{R}^{48}$, touch $o_\text{T} \in \mathbb{R}^{92}$, and object state $o_\text{O} \in \mathbb{R}^{13}$ values to infer action a, i.e. the next joint positions to be executed by the simulator environment. An object's state o_O comprises the pose (position o_{O_P} and quaternions o_{O_Q}) and velocities (linear o_{O_L} and angular o_{O_A}).

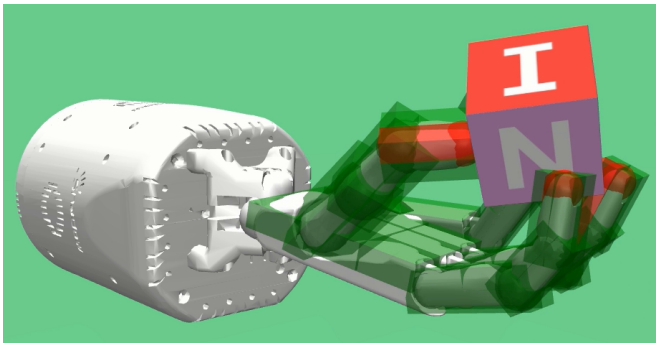

Fig. 1: MuJoCo simulation of a Shadow-Hand for dexterous in-hand manipulation with 92 touch sensors from [1]. Touch sensor sites, activated by the contact with the cube, are highlighted in red while the inactive sites remain green.

Since o_O is susceptible to drop-outs due to slower update frequency or occlusion, only partial observations $\tilde{o}$ can be made during drop-out. To reconstruct the full observation, we apply a deep neuronal network f as follows:

$$o_{t+1,\text{O}} = f\left(a_t, h\left(o_t, \tilde{o}_{t+1}\right)\right) \text{ with } h\left(o_t, \tilde{o}_{t+1}\right) = \left(o_{\text{O,t}}, o_{\text{P,t}}, o_{\text{P,t+1}}, \tanh o_{\text{T,t}}, \tanh o_{\text{T,t+1}}, \tanh \frac{o_{\text{T,t}} - o_{\text{T,t+1}}}{2}\right).$$

f is a common feed-forward network as we assume the change of the object's state to be a first-oder Markovian process wrt. pose and velocity. In comparison to [1] we normalized the touch values from $[0., \infty)$ to $[0., 1.)$ using the tangens hyperbolicus (tanh).

We achieved best results by configuring f as follows: $f = (f_\text{P}, f_\text{Q}, f_\text{L}, f_\text{A})$ consists out of four deep networks to infer o_{O_P}, o_{O_Q}, o_{O_L}, and o_{O_A} independently; each network has three hidden layers with 512 neurones and ReLU activations; f_P, f_L, and f_A have linear output layer while f_Q has tanh activation with L_2-normalization; f_P, f_L, and f_A are trained using mean-squared-error loss while we applied log cosine-similarity loss to f_Q as in [2]; Adam is used as optimizer with learning rate=.001, β_1=.9, β_2=.999, and batch-size=128.

We recorded 100 steps from 19001 epochs with an up-

[†]Bielefeld University, Cluster of Excellence Cognitive Interaction Technologies, Inspiration 1, 33619 Bielefeld, Germany
[‡]JL is with the Australian Centre for Robotic Vision, Queensland University of Technology, Brisbane, Australia
[*]{tkorthals, amelnik}@cit-ec.uni-bielefeld.de
This research was supported by the Ministry of Economy, Innovation, Digitization and Energy of the State of North Rhine-Westphalia (MWIDE) within the Leading-Edge Cluster 'Intelligent Technical Systems OstWestfalenLippe (it's OWL)', supervised by Projektträger Jülich (PtJ), the Federal Ministry of Education and Research (57388272), and by 'CITEC' (EXC 277) at Bielefeld University which is funded by the German Research Foundation (DFG). The responsibility for the content of this publication lies with the author.

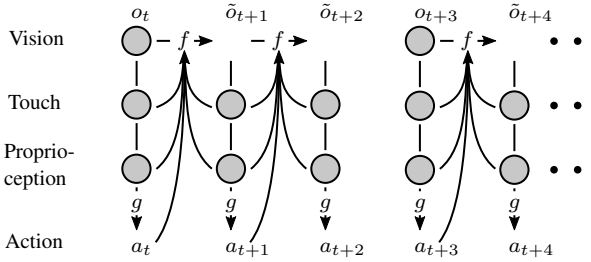

Fig. 2: Reconstruction of partial observations $\tilde{o}$ from past and current observations plus action. The approximation f predicts the missing vision based information. The policy network g infers the next action a_t based on the restored observations.

Err.	Drop	naïve	baseline	w/o	w/
pos.	1	.013	.013	.0033	**.0032**
	2	.018	.0131	.0042	**.0039**
	5	.026	.0132	.0061	**.0056**
quat.	1	.00209	.00209	.00042	**.00037**
	2	.0039	.00210	.00075	**.00067**
	5	.0091	.00213	.0019	**.0017**
lin. v.	1	.45	.45	.063	**.058**
	2	.51	.450	.065	**.060**
	5	.56	.452	.069	**.064**
ang. v.	1	4.27	4.27	1.19	**1.13**
	2	4.49	4.28	1.21	**1.15**
	5	4.70	4.32	1.26	**1.19**

TABLE I: Mean error metric for the reconstruction quantities over number of drop-outs.

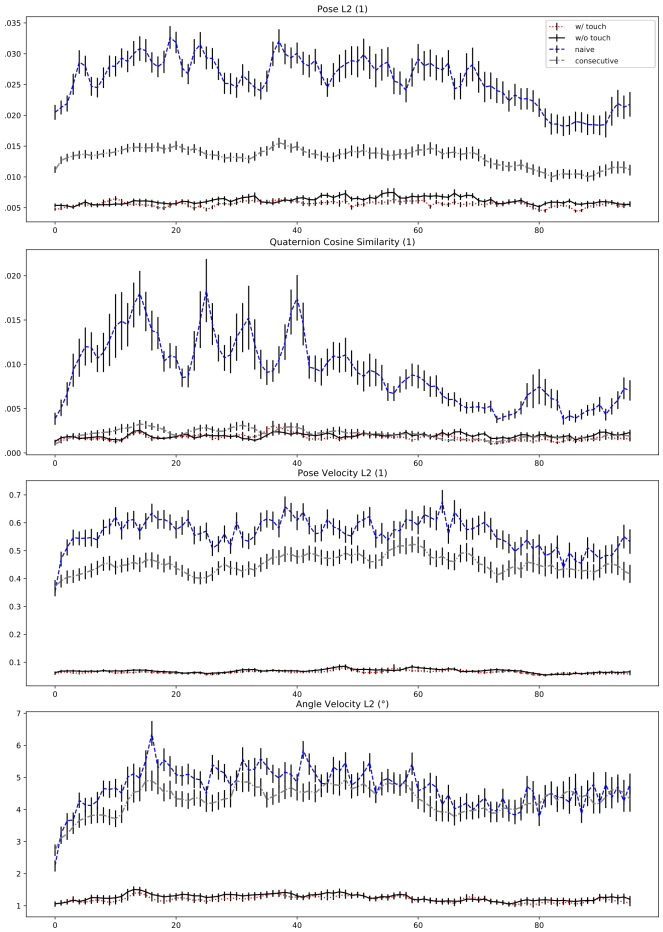

Fig. 3: Error of 6-DoF vision based information pose (position, rotation) and velocities (linear, angle). Average error with 10 % standard deviation over 30 epochs for predicting five steps. Experiment depicts prediction without (w/o), with (w/), and normalized (tanh w/)) touch versus naïve (naive) and baseline (consecutive) error.

date rate of 25 Hz of the trained agent from [1] without sensory drop-out via the MuJoCo Shadow-Hand simulation [3] manipulating a cube (c.f. Fig. 1). The data set tuples for training f is constructed with $(a_t, h\,(o_t, \tilde{o}_{t+1}))$ as input and $o_{t+1,\mathrm{O}}$ as output for each recording, which results in $99 \cdot 19001 = 1881099$ samples from which 98 % were used for training and 2 % for validation.

II. RESULTS

We trained the network for reconstruction with (w/) and without (w/o) touch information and evaluated the corresponding error metrics as shown in Table I. The tanh normalization of touch values led to slightly better results in comparison to un-normalized values which results we leave out for brevity. We additionally evaluated the error for a naïve and baseline case. The naïve approach just freezes the last received, vision based, perception over all ongoing steps where vision is not available. The baseline serves as a hypothetical comparison where the consecutive errors of all vision based, step-wise ground-truth values are evaluated (therefore, naïve and baseline are equal for Drop=1).

Table I shows that the network performs always better in reconstructing ongoing vision based information with touch in comparison without touch. Additionally, the error per step over 30 epochs is shown in Fig. 3 for Drop=5. The trend of the error curves, which is higher for the first steps and decays until the end, reveal the behaviour of the policy network which tends to perform more rapid movement in the beginning. Therefore, consecutive errors are much bigger. However, the network reconstructs the object poses via haptic and proprioceptive information with almost constant quality. The presented approach substantially improves the estimation of poses of a manipulated object when a visual drop-out takes place.

REFERENCES

[1] A. Melnik, L. Lach, M. Plappert, T. Korthals, R. Haschke, and H. Ritter, "Increasing Sample Efficiency and Performance in Reinforcement Learning for In-Hand Manipulation Tasks by Touch Sensing," in *2019 IEEE International Conference on Intelligent Robots and Systems (SUBMITTED)*, Macao, China, 2019. [Online]. Available: https://rebrand.ly/Melnik2019TouchSensors

[2] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," *CoRR*, vol. abs/1612.05424, 2016. [Online]. Available: http://arxiv.org/abs/1612.05424

[3] M. Plappert, M. Andrychowicz, A. Ray, B. McGrew, B. Baker, G. Powell, J. Schneider, J. Tobin, M. Chociej, P. Welinder, V. Kumar, and W. Zaremba, "Multi-goal reinforcement learning: Challenging robotics environments and request for research," 2018.