Attention-based Robot Learning of Haptic Interaction

Alexandra Moringen⋆, Sascha Fleer⋆, Guillaume Walck⋆, and Helge Ritter

Neuroinformatics Group, Bielefeld University `abarch@techfak.uni-bielefeld.de`

Abstract. Haptic interaction involved in almost any physical interaction with the environment performed by humans is a highly sophisticated and to a large extent a computationally unmodelled process. Unlike humans, who seamlessly handle a complex mixture of haptic features and profit from their integration over space and time, even the most advanced robots are strongly constrained in performing contact-rich interaction tasks. In this work we approach the described problem by demonstrating the success of our online haptic interaction learning approach on an example task: haptic identification of four unknown objects. Building upon our previous work performed with a floating haptic sensor array, here we show functionality of our approach within a fully-fledged robot simulation. To this end, we utilize the haptic attention model (HAM), a meta-controller neural network architecture trained with reinforcement learning. HAM is able to learn to optimally parameterize a sequence of so-called haptic glances, primitive actions of haptic control derived from elementary human haptic interaction. By coupling a simulated KUKA robot arm with the haptic attention model, we pursue to mimic the functionality of a finger.
Our modeling strategy allowed us to arrive at a tactile reinforcement learning architecture and characterize some of its advantages. Owing to a rudimentary experimental setting and an easy acquisition of simulated data, we believe our approach to be particularly useful for both time-efficient robot training and a flexible algorithm prototyping.

Keywords: Haptic interaction in 3D · Reinforcement learning · Haptic attention · Robot control.

1 Introduction

Most activities such as sports, high-precision dexterous handling of tools or playing musical instruments take place through *haptic interaction* with the 3D environment. Under *haptic interaction* we understand a physical interaction with objects established by active touch. A number of examples demonstrate the importance of haptics in order to perform a dexterous task successfully. The most famous example is the experiment in which a study participant was asked to light up a match with anesthetized fingers [4] and encounters extreme difficulties doing so. Unlike humans, who, after years of developmental process [15], seamlessly handle a complex mixture of haptic features [9] and profit from their integration over space and time [6], even the most advanced robots are strongly constrained in performing contact-rich interaction tasks [2]. This is due to several reasons. In contrast to other fields such as computer vision, encompassing haptic

---

⋆ The authors contributed equally to this work

interaction benchmark sets do not exist yet. Many questions about a general approach to haptic interaction modeling are still unanswered, e.g.: How to represent multimodal haptic characteristics of the explored object, such as rigidity, temperature, or texture? How to integrate over space and time, and how to organize the corresponding haptic memory? How to perform efficient control that – in turn – produces the most suitable data to ensure a successful task progress? How to represent a primitive haptic action? Overall, it remains an open question, how to enable robots to perform haptic interaction with the 3D environment, a skill that should finally allow them to e.g. achieve a human or even superhuman level of dexterity, compared to results achieved on the basis of computer vision alone [3]. In this work we propose to advance in this direction with our rudimentary tactile reinforcement learning infrastructure which we believe to be useful for a versatile development of the robot dexterous manipulation.

Framework. Because ideally all the above issues should be addressed within one framework, the contribution of this work is a systematic approach integrating four highly modular components: 1) primitive haptic actions, haptic glances (HGs) [1] 2) a haptic attention model (HAM), illustrated in Figure 1, that performs an optimal sequence of primitive haptic actions given a high-level goal specification [1], 3) a modular world model MHSB [8, 7, 5] that can serve as a platform for a given task specification in a three-dimensional space, 4) a physics-driven simulation environment Gazebo incorporating all experimental components, the robot arm equipped with a tactile sensor array and the 3D objects (see Fig. 2). We show how the above framework enables us to perform haptic interaction learning in simulation by successfully solving an object classification task.

Contributions. The major contribution of the proposed work, compared to previous approaches to perform haptic interaction with robots (e.g. [12, 14, 11]), is the absolutely minimal amount of hard-coded inputs, hand-crafted preprocessing or other prior knowledge, e.g. human demonstrations, necessary for a successful task performance. The skill represented by a control policy is learned from scratch, and the only input consists of a specification of a primitive haptic action type, and a reward for a successful task execution. In our previous work [1], we have developed a learning architecture that is able to shape the exploration process of a floating tactile sensor through directing its attention on salient tactile features of objects. As a first step towards real robot integration, the present paper is now porting the designed learning architecture into a realistic robot simulation. In this work we also show that our model optimized with a limited cached data set, generalizes well to a performance with new data acquired online. Due to the above results, we believe that our work may be particularly suitable as a foundation for learning of more complex tasks, such as e.g. assembly or search. We encourage you to watch the video that is provided within the supplementary material for a quick overview of the presented paper.

2   Robot control and data acquisition

Even if the proposed method is generic and can be applied in many different scenarios, we chose to demonstrate the applicability of the concept with a setup designed to acquire tactile signals with a tactile sensor array mounted on a robot arm, while exploring
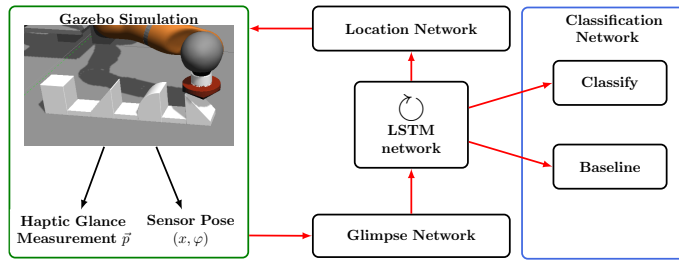
Fig. 1: Illustration of HAM. Generated tactile data *p* of the simulation is combined with the position and the orientation of the sensor within the *glimpse network*. The generated features are then processed through the *LSTM network* in order to either classify the given object, employing the *classification network* or to generate a new pose for the sensor by utilizing the *location network*. The pose is employed to perform a further haptic glance within the simulation.
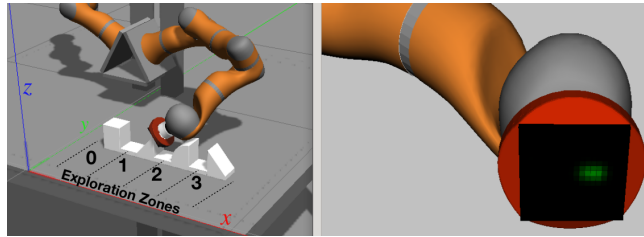

Fig. 2: Simulation of the KUKA LWR4 robot arm (7 degrees of freedom) with the Myrmex sensor array attached to the end-effector. Left: the robot performs a haptic glance by establishing contact with the object of $\sim$10 cm^2 footprint placed in exploration zone 1. Right: corresponding visualization of the resulting tactile measurement. A simulated version of the sensor was created for Gazebo that mimics the contact distribution over the array through the use of a mixture of Gaussians around each contact point, weighted according to the local contact force.

a stimulus. Our inspiration is therefore the functionality of a finger performing haptic interaction with fingertip-sized objects, similar to [14].

Exploration Zones. In order to learn an exploration policy that is independent of the object's pose within the global coordinate system, we introduce *exploration zones* (see again Figure 2). Each exploration zone is a pre-defined region $\sim$20 cm wide in front of the robot, in which the $\sim$10 cm wide objects are centred for exploration. The exploration zones define their own local reference frame, with normalized coordinates to cover the range $[-1, 1]$ around the origin. After specification of the exploration zone, two out of six pose parameters of the tactile sensor can be modified by the HAM: the position $x \in [-1, 1]$ along the *x*-axis within the coordinate frame of the corresponding exploration zone, and the orientation angle $\varphi \in [-0.3\pi, 0.3\pi]$ around the *y*-axis.

Robot arm and control. The robotic setup consist of a KUKA LRW4 robot arm with 7 degrees of freedom ensuring a range of motion similar to a human arm. Its end-effector is equipped with the tactile sensor Myrmex mounted on an ATI force-torque sensor as shown in Fig. 2. The purpose of the setup is to explore the stimuli with the sensing surface in a safe manner. Robotic interactions with the environment always require great care to avoid damage due to unintended high contact forces. Therefore,

unplanned contacts are usually not desired. Since the exploration procedure is guided by the learning system, the various sensor poses executed on the robot are not known in advance. Moreover, for more realism, the shapes to explore are also unknown but contained in a $\sim$10 cm^2 footprint, which forbids any planning for obstacle avoidance. Hence, the robot arm should move and rely on events to react accordingly when touching the environment. Tactile events are not sufficient to stop motion, because the contact could also occur on non-sensorized surfaces of the robot. To complement the tactile events, two other events are taken into account. First the force-torque sensor mounted between the tactile sensor and the last joint of the arm triggers an event when a force threshold is reached. The force is induced by the contact between the environment and any part of the tactile sensor, even on the non-pressure reactive surfaces of the assembly (dark orange part in Fig. 2). The force threshold is higher than the minimum force needed to trigger the tactile array to ensure pressure data can be acquired before stopping motion due to contact forces. Secondly, an event is generated when other parts of the robot touch the environment. We rely on the joint-impedance control mode of the robot, that permits to select the stiffness of the arm when reaching a certain posture. The "softer" the stiffness is set, the larger can the deviation be relative to the desired posture. This allows to execute a motion that penetrates or collides with an obstacle (here the stimuli), but exerts small forces on contact without crushing the sensor or damaging the arm limbs. The deviation between the actual pose and the desired one can be monitored and an event is triggered if the deviation is too large, meaning the robot was stopped by an obstacle. To summarize, the motion is stopped either by a tactile event, which is a successful data acquisition, or by a too high force on contact between the end-effector and the environment, or by a too large deviation between the desired joint target and the actual in case of a contact with other robot body parts, both latter events being considered as a failed data acquisition. As a first step, the whole robotic system was recreated in simulation, using Gazebo and a simulated LWR robot controller providing impedance control in joint space. The real-time control loop consisting of a Cartesian controller and of a Cartesian trajectory controller (interpolating motions and monitoring deviation), is exactly the same for the real-world robot, and permits to validate the safety mechanism and the algorithms in the virtual environment first. Due to time constraints, the data used is currently from simulation only, which could be acquired rapidly in unattended mode.

Haptic Glances and Haptic Glance Controller. Haptic Glance Controller (HGC) is the interface between HAM and the robot simulation. The robot is controlled by the HGC via a state machine receiving the target pose for each individual HG from the HAM as depicted in Fig. 3. HGC requests new exploration poses which the state machine executes following a sequence of three states. In *To Pos* the sensor is moved to the pose (x, φ) above the objects, while z remains at the constant pre-defined level. Then, the *Go Down* state queries a slow downwards motion, while monitoring the high-force, deviation and tactile pressure events. On any of the events, the state-machine switches to the *Go Up* state, moving the sensor away from the object. In the case of a tactile event, the data is transmitted back to the HAM, completing one haptic glance.

A HG in this work is implemented as a movement downwards towards the object, while sustaining a given pose, until a contact is established. Each HG that is executed
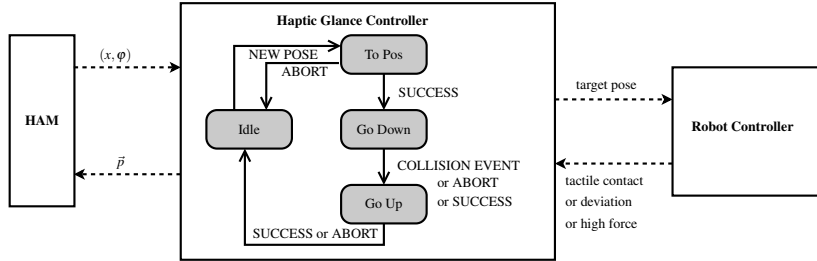
Fig. 3: Haptic Glance Controller receives the pose from the HAM and forwards it to the state machine employed for the robot control. During haptic interaction the robot switches between three main states, *To Pos* (move to a given pose), *Go Down* (establish contact by going down until collision is detected), and *Go Up* (go up after contact has been established).

within the Gazebo simulation is represented by the pose (x, φ) of the tactile sensor within the associated exploration zone, but converted to a 6D coordinate for the robot end-effector to reach. Before the execution of a haptic glance the sensor is placed at the specified pose, with the height of the sensor above the given exploration zone being predefined manually. In order to establish the contact, the sensor is moved down along the z-axis, and a corresponding pressure vector p is recorded once any sensitive cell of the sensor reaches a pre-defined threshold.

3   Experiments and Results

To avoid repeated acquisitions of the same or very similar haptic glances in simulation, and to enable an efficient evaluation of the model hyperparameters, we create a dataset of pose-pressure tuples. To this end, we tessellate the whole location-orientation space that can be accessed by the sensor and generate a cache of haptic glances that are then stored in a dataset for learning. Hence, our experiment is split in two parts. As in our previous work, we train our model by using the dataset until a high accuracy is learned. We then show that our trained model is able to generalize beyond the recorded data, we utilize the best model for all trained number of glances and test their performance by classifying the four objects within the fully-fledged simulation online.

Dataset. The dataset is generated by recording tuples $d_o = (p, x, \varphi)$ of the normalized pressure data p, together with the corresponding normalized location x and orientation φ of the sensor. For the data to be independent of the object global pose, the location data $x \in [-1, 1]$ is given within the local coordinate frame of the exploration zone. After reaching the corresponding exploration zone with the robot, the recording of data points starts at $x = -1$ with the orientation $\varphi = -0.3\pi$. After covering 41 discrete orientations φ with a step size $\Delta_\varphi = \pi \cdot 0.05$, the location is incremented by $\Delta_x = 0.05$ and the recording of 41 orientations starts anew, until 41 locations are covered. Leading to 41×41 pre-recordings per object and to a full dataset of 6724 data points. During training, the model generates location-orientation pairs (x, φ) for which the corresponding pressure vector p is directly extracted from the dataset at data point d_o that best matches (x, φ), instead of re-measuring the pressure vector in simulation.

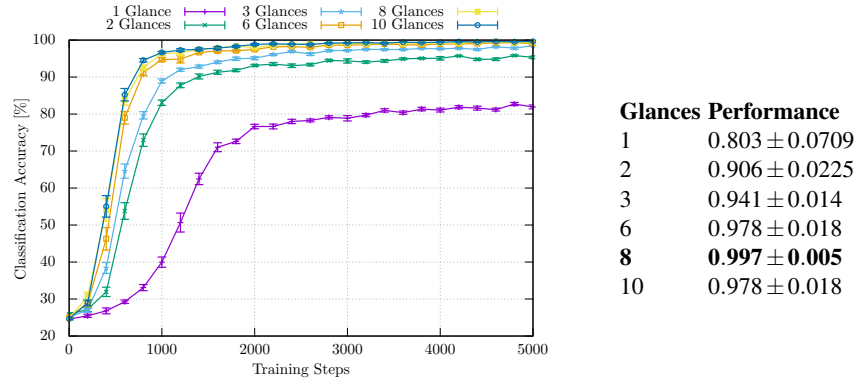| Glances	Performance
1	0.803 ± 0.0709
2	0.906 ± 0.0225
3	0.941 ± 0.014
6	0.978 ± 0.018
8	**0.997 ± 0.005**
10	0.978 ± 0.018

Fig. 4: Evaluation of the models learning performance when trained on the pre-recorded dataset. The speed and efficiency of the learning is depending on the number of executed glances per classification step. The table on the right lists the measured classification performance, recorded in simulation using the best model, pre-trained on the recorded dataset. The results are averaged over 4 trials.

Model Training. The pre-recorded dataset is used to train the designed model on a different number of glances for 5000 training steps. For evaluation, the training is stopped after a predefined step interval. The current policy is then evaluated on 100 test batches in which all of the four objects have to be identified an equal number of times[1]. Even for such a small dataset, the designed model is able to identify the different objects with a nearly perfect score of $\approx 100\%$ for 10 glances (see Figure 4).

Testing the model on the simulated robot arm online. After successfully training a model that is able to classify the four objects with high accuracy while only using the limited data of the pre-recorded dataset, the learned model is now tested within simulation. For testing the quality, every object is presented 20 times for classification within 4 distinct trials. The results are listed in the table in Figure 4. Even the use of one single haptic glance per object leads to an accuracy of more than 80%. Again, the classification performance is increasing when more glances are used. While adding a second glance increases the success rate about 10%, the third one only adds a gain of $\approx 4\%$ and further increases with every additional glance added. Nevertheless, an accuracy of more than 99% can be reached for this simple task when 8 glances are used. For 10 glances, the accuracy is slightly dropping to about 98%.

4   Discussion

A physics-driven control of the robot arm employed in this work vs. the position-based control performed without gravity in our previous work, resulted in a more realistic and less noisy haptic data. As a result we could demonstrate a higher reliability and a faster convergence of the trained model when applied to a simulated robot. This is a good indicator that our research will also lead to fruitful results when applied on a

---

[1] As a batch size of 64 is used for training, every object has to be identified 1600 times during the performance evaluation.

real robot platform. Furthermore, this work specifically gives implementation details on the robotic setup, as this aspect is inherently difficult. Its emphasis is on the safety mechanisms required to gather data in an unknown environment without risking major robot or sensor failures. Even with those safety measures, preparing the reduced 41×41 data set on a real robot still requires attendance, while simulation data permitted to extract first promising results unattended.

Additionally the work explored model learning (training) and exploration execution (testing) on different data sets, pre-recorded and live set acquired online in the simulation environment, respectively. The time factor is a huge problem in employing deep reinforcement learning in robotics. Therefore, the usage of a pre-recorded data without new generation of data in each test iteration may be a promising methodology for a development of algorithms and useful for transition to real-world data sets. Importantly, the usefulness of the pre-recorded sets remains to be tested w.r.t. its advantages for the transition to the real-world performance.

Altogether, we believe that this work may serve as a foundation that brings the known framework of active vision and glances to a different modality with haptic glances. It integrates haptic glances in RL and performs learning of haptic interaction based on physics-driven robot arm control, leading to faster convergence and increased reliability of the resulting model. This opens different possible above-mentioned research directions.

5 Conclusion and Future Work

This work presents an approach for teaching a simulated robot equipped with a tactile sensor, how to classify four objects from data gathered with haptic glances at one or more sensor poses. In order to answer the question how these poses should be selected in an optimal way, we adapted the *haptic attention model*. This model enables us to learn efficient haptic interaction by integrating over the time-series of acquired tactile sensor data while simultaneously improving the current policy. In order to enable fast hyperparameter optimization and avoid multiple calculations of the same data in simulation, we have pre-recorded a dataset of haptic glances (p, x, φ). First tested on this small set, our approach reaches nearly optimal classification performance. With the goal to evaluate the generalizability we then exploited the same model for performing the classification task within the simulation environment online. Despite a relatively small training set compared to the number of trainable variables, the network shows good generalization performance as demonstrated by the results achieved in the online simulation. This is in line with findings in literature, that state that a large overparameterization does not necessarily lead to overfitting [10].

Training of the model solely on a pre-recorded data set might not be enough for more complicated tasks on the one hand. On the other hand, a full training even within the online simulation is likely to be time consuming. Therefore further approaches need to be investigated. One possibility is to use a *transfer learning* approach [13] by first training the model on a pre-recorded dataset and then adding refinement to the learned policy by training the same model for a smaller number of training steps directly on the simulated robot setup. A next step would be to make the transition from the simulated

robot to a real-world setup, using the proposed safety mechanisms, but performing un-planned poses still requires attendance. Hence, a reasonable intermediate step would be again to pre-record a dataset with predictable safe poses. Training with this real-world dataset should show how well the model can deal with the noise within the data that is inevitably present when working with a real robotic setup.

References

1. Fleer, S., Moringen, A., Klatzky, R.L., Ritter, H.: Learning efficient haptic shape exploration with a rigid tactile sensor array. PLOS ONE **15**(1), 1–22 (01 2020). https://doi.org/10.1371/journal.pone.0226880, https://doi.org/10.1371/journal.pone.0226880
2. Lee, M.A., Zhu, Y., Srinivasan, K., Shah, P., Savarese, S., Fei-Fei, L., Garg, A., Bohg, J.: Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks. arxiv (2019)
3. Levine, S., Finn, C., Darrell, T., Abbeel, P.: End-to-end training of deep visuomotor policies. J. Mach. Learn. Res. **17**(1), 1334–1373 (Jan 2016), http://dl.acm.org/citation.cfm?id=2946645.2946684
4. Match lighting experiment. https://www.youtube.com/watch?v=0LfJ3M3Kn80
5. Modular haptic stimulus board. https://www.youtube.com/watch?v=CftpCCrIAuw
6. Morash, V.S., Pensky, A.E.C., Miele, J.A.: Effects of using multiple hands and fingers on haptic performance. Perception **42**(7), 759–777 (2013)
7. Moringen, A., Haschke, R., Ritter, H.: Search procedures during haptic search in an unstructured 3d display. In: IEEE Haptics Symposium (2016)
8. Moringen, A., Aswolinskij, W., Buescher, G., Walck, G., Haschke, R., Ritter, H.: Modeling target-distractor discrimination for haptic search in a 3d environment (2018), bioRob
9. Panday, V., Tiest, W.M.B., Kappers, A.M.L.: Bimanual integration of position and curvature in haptic perception. IEEE Transactions on Haptics **6**(3), 285–295 (July 2013). https://doi.org/10.1109/TOH.2013.8
10. Poggio, T.A., Kawaguchi, K., Liao, Q., Miranda, B., Rosasco, L., Boix, X., Hidary, J., Mhaskar, H.: Theory of deep learning III: explaining the non-overfitting puzzle. CoRR **abs/1801.00173** (2018), http://arxiv.org/abs/1801.00173
11. Shenoi, A.A., Bhattacharjee, T., Kemp, C.C.: A crf that combines touch and vision for haptic mapping. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 2255–2262 (Oct 2016). https://doi.org/10.1109/IROS.2016.7759353
12. Sommer, N., Billard, A.: Multi-contact haptic exploration and grasping with tactile sensors. Robotics and Autonomous Systems **85**, 48 – 61 (2016). https://doi.org/https://doi.org/10.1016/j.robot.2016.08.007, http://www.sciencedirect.com/science/article/pii/S0921889016301610
13. Taylor, M.E., Stone, P.: An Introduction to Intertask Transfer for Reinforcement Learning. AI Magazine **32**(1), 15 (2011). https://doi.org/10.1609/aimag.v32i1.2329, http://www.aaai.org/ojs/index.php/aimagazine/article/view/2329
14. Tian, S., Ebert, F., Jayaraman, D., Mudigonda, M., Finn, C., Calandra, R., Levine, S.: Manipulation by feel: Touch-based control with deep predictive models. arxiv (2019)
15. Withagen, A., Kappers, A.M.L., Vervloed, M.P.J., Knoors, H., Verhoeven, L.: The use of exploratory procedures by blind and sighted adults and children. Attention, Perception, & Psychophysics **75**(7), 1451–1464 (Oct 2013). https://doi.org/10.3758/s13414-013-0479-0, https://doi.org/10.3758/s13414-013-0479-0