Using Haptics to Extract Object Shape from Rotational Manipulations

Claudius Strub[1,3], Florentin Wörgötter[1], Helge Ritter[2], and Yulia Sandamirskaya[3]

Abstract— Increasingly widespread available haptic sensors mounted on articulated hands offer new sensory channels that can complement shape extraction from vision to enable a more robust handling of objects in cases when vision is restricted or even unavailable. However, to estimate object shape from haptic interaction data is a difficult challenge due to the complexity of the contact interaction between the movable object and sensor surfaces, leading to a coupled estimation problem of shape and object pose. While for vision efficient solutions to the underlying SLAM problem are known, the available information is much sparser in the tactile case, posing great difficulties for a straightforward adoption of standard SLAM algorithms. In the present paper, we thus explore whether a biologically inspired model based on dynamic neural fields can offer a route towards a practical algorithm for tactile SLAM. Our study is focused on a restricted scenario where a two-fingered robot hand manipulates an n-gon with a fixed rotational axis. We demonstrate that our model can accumulate shape information from reasonably short interaction sequences and autonomously build a representation despite significant ambiguity of the tactile data due to the rotational periodicity of the object. We conclude that the presented framework may be a suitable basis to solve the tactile SLAM problem also in more general settings which will be the focus of subsequent work.

I. INTRODUCTION

Grasping and object manipulation are the core capabilities of robots in assistance and production scenarios. Especially in the assistance scenario, the ability to learn to manipulate unknown objects is desirable, since the system cannot rely on a database of shape parameters of all objects which the robot may encounter. In such setting, the knowledge about the object shape has to be obtained using the on-board sensors of the robot. Typically, computer vision is used to estimate the object boundaries for grasping and object manipulation. Depending on the visual appearance of the object, its shape estimation with vision may be poor or unavailable. Since haptic sensors are available nowadays on most robotic hands, haptics could be used as a complementary mechanism to access object shape. Haptic sensors are directly coupled to the effector and thus provide information, which is directly relevant for manual manipulations of objects.

Despite its potential for shape estimation, the tactile feedback is mostly considered to determine whether the

manipulator has contact to the object and to control the force of the grasp [24], [25]. To the contrary, humans and other primates use haptics in a much richer way, e.g. to determine pose, shape, texture, or weight of the object, even in the absence of visual feedback [15], [32]. This raises the question, whether haptic information may be used more extensively to access object shape.

In this paper, we explore haptic learning in a simple setting, which allows to access some principled problems with this approach and suggest a model, which allows to cope with these challenges. Understanding the mechanisms, underlying haptic learning, would lead to a modality specific information which could be fused and complemented with those of other modalities, e.g. vision.

One constraint, which we set in our work is that the system has to incrementally build an object representation in its interaction with this object. We would like to contrast such incremental, online learning to batch learning, where all the past experiences are stored in a raw form and are used to build the model later. In an online learning scenario, the system starts with a simple uncalibrated manipulation behavior, permitting controlled periodic contacts with the object and improves this behavior over time by building an object representation. The latter improves behavior by enabling the agent to predict action outcomes and thus to make goal directed modifications. In this case, the data acquisition, training, and exploitation phases are highly interwoven.

Moreover, we are interested in a learning mechanism, which is not dependent on a specific behavior, e.g. a predefined series of specific gasps of a rigidly mounted object. Indeed, most state of the art approaches to haptic learning use one of two simplifications in the haptic learning process: the first class of systems uses rigidly mounted objects for learning their shape and geometry, e.g. [20], [18], [7], the second class of systems uses haptics to localize objects, the shape of which is assumed to be known [10], [23], [5]. If the geometry is initially unknown and the pose of the object changes during the learning phase, the haptic learning problem becomes equivalent to the well-known simultaneous localization and mapping (SLAM) problem in navigation.

Instead of localizing the robot with respect to the environment, the task here is to localize an object with respect to the robot. In this case, the pose of the object has to be estimated and continuously tracked during online manipulations based on the perceived features at contact points. Small errors in this pose estimate accumulate over time and lead to a drift in the mapping of the object geometry, which calls for an error correction mechanism to prevent drift in the estimate. In the field of robotic mobile navigation, multiple approaches

are known to solve the SLAM problem, for a review see [8], [2]. There have also been several biologically motivated approaches to the problem, e.g. [19], [21], [6].

Tactile SLAM has several differences compared to its navigation analogous, however. In tactile SLAM, the sensor information is only present during periods of object contact in contrast to the typical continuously available information of distance sensors (e.g. sonar, infra-red, or laser). This leads to only very sparsely distributed information in space, comparable to solving SLAM in navigation by only using the bumpers of a robot. Furthermore, the resulting sensor data typically does not have spatially distinct features, as surface curvature, edges, and texture are often ambiguous in space. Hence, the capabilities of computing, tracking, detecting and matching unique or salient features, which serve as landmarks, are very restricted. This renders most traditional solutions to the SLAM problem unsuitable for a purely haptics based setup, although modifications for particle filters have recently been proposed in order to cope with spatial sparsity of contact informations [16]. In [9] the tactile SLAM problem is approached by rasterizing the environment into a binary grid-map encoding 'empty' or 'taken' and then including this map into the particle filter. Tactile measurements are then incorporated by using assumptions with respect to the environment. However, this approach lacks the necessary precision for predicting tactile input patterns and is additionally computationally intractable for a reasonable map resolution. In [35] a Bayesian approach to the tactile SLAM problem is suggested, although substantial pre-knowledge is needed even for simple simulation experiments (e.g. a "localization sensor", known dimensions and mass of the object).

The model, which we propose here is inspired by biological processing of haptic pathways, in that the preprocessing of information leads to biologically plausible features and the performed computational methods respect neural processing mechanisms. This allows us to potentially use and refine the same model in accounting for human behavioral data on haptic learning, if these become available. One principle which we take from biology is separation of shape representation from the representation of the object pose, i.e. an explicit object representation. In humans there is evidence for this from developmental psychology studies, e.g. [31], [33]. Neurobiological work also indicates that object representations are invariant to the pose, which holds for vision [13] as well as for haptics [11]. This motivates learning of the explicit object shape representation in our model. Insights into the processing of haptic information show similar mechanisms to those known from the visual pathway [4], [11], [32]. Additionally, vision and haptic pathways for constructing an object representation are highly interleaved, multi sensory processing and integration takes place at a variety of stages [14], [12], [32], [22]. This legitimates the application of algorithms form the field of biologically inspired computer vision research [17] to tactile sensor data.

As a first step towards solving the fully-fledged haptic SLAM problem, a simple setup containing objects with one degree of freedom in the orientation, is investigated. In this experimental setting, we use two fingers of a robotic hand to rotate different objects and incrementally build the representation of the object shape during the manipulation. In particular, the available features from haptic data are locally matched in the rotational dimension in order to retain a consistent representation of the object shape; errors in this representation are detected and corrected. This demonstrates the principled ability of the model to build these representations in an online haptic learning process.

II. Overview of the Setup

This section gives an overview of the developed architecture for using haptics to build object shape representation. A scheme of the architecture is shown in Fig. 1. The neural-dynamic model, which implements the central parts of this architecture is further explained in Section III, and the setup used for evaluation of the model is described in Section IV.

A. The Manipulation Behavior

In the upper part of Fig. 1, the behavioral loop is depicted, which amounts to a reactive behavior controlling a robot manipulator, providing tactile and proprioceptive sensory feedback.

In particular, the forward kinematics of the fingers was solved analytically and the desired rotation behavior was learned on a set of kinesthetic teaching examples: objects of different sizes were manually rotated with the robot hand while recording the joint angles. The computation of the desired mapping was done via Principle Component Analysis (PCA) on the recorded training data in joint angle space. Only the first two PCs were used as parameters for the rotation behavior: the first PC corresponds to the rotation angle of the fingers and the second PC corresponds to the grasp diameter.

The rotation parameter was constantly increased until one of the tactile sensor activation blobs reached the spatial
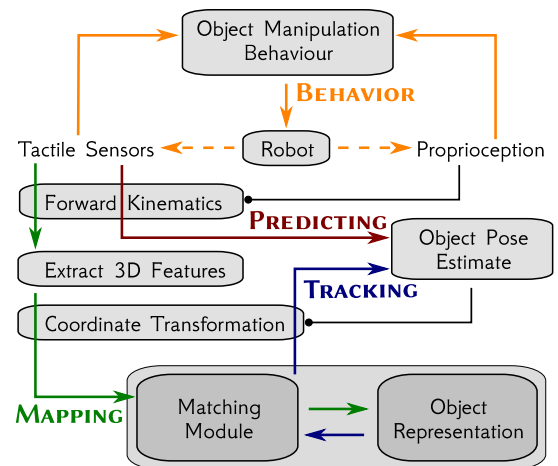

Fig. 1. Overview of the general architecture: black lines with a dot at the end indicate a parametrization relationship. Arrows pointing to boxes indicate inputs for processing in the box, and arrows passing through boxes indicate a parametrized transformation of the information through the box.

boundary of the sensor surface. When the border of the sensor surface is reached, the grasp is released via the grasp diameter parameter and the rotation parameter is reset to a point-symmetric position (see Fig. 3, left). Finally, the grasp is closed and the rotation movement begins all over again. While the rotation is performed, the grasp diameter is continuously controlled for, such that the tactile sensors report a desired pressure level and do not loose contact with the object. This behavior was implemented outside the neural-dynamic framework, but may be generated using the neural dynamics approach, as described in [26]. The object was supported by the planar table-surface all the time, hence no force closure or other stability criteria had to be taken into account.

The remainder of Fig. 1 shows the steps necessary to learn a model of the object shape based on the acquired haptic data.

B. The Problem of Predicting the Object Pose

We assume that the kinematics of the hand are known, thus a forward model is able to give a prediction of the manipulated object pose, based on proprioception and tactile contacts. The difference between the positions of two consecutive tactile contact measurements in time allows to predict the change in rotation, and if there is a pose estimate, also the translational change of the object.

Due to only having two joints per finger, there are not only forces orthogonal to the object surface, but also tangential components. These lead to an uncontrolled movement when the object is released, which cannot be detected nor prevented in the proposed setup. Together with the unintended slight shift and rotation of the object when the grasp is closed again, these are the main sources of noise in the orientation estimate. In general, these should be systematic and indeed there is a strong tendency of systematically underestimating the object rotation. In contrast, slippage was nearly negligible, due to a very "grippy" texture on the object surface. In Section V, we demonstrate how the model compensates for these errors.

C. Computing Haptic Features

For mapping the object shape, features are extracted from the tactile sensors and transformed to external three dimensional coordinate system, combining the tactile sensor information with joint angles, based on the known kinematics. The proposed model is inspired by neural processing of haptic pathways and therefore operates on tactile features which are similar to features known to play a role in visual pathways. These features include: zero-, 1st, and 2nd order moments, which correspond to position, orientation and curvature of the tactile contact, respectively.

In order to compute these features during manipulation, the joint angles, centroids and covariances of the tactile pressure patterns (via OpenCV CvBlob detection) are stored in tactile coordinates. Then the three-dimensional position of the contact points and the corresponding normal vector of the tactile sensor surface is computed using the forward kinematics. In this particular state of the project, the object

motion is restricted to one degree of freedom: rotation along it's z-axis. Therefore, the third dimension of the features (i.e. height / z-axis) is currently irrelevant and thus omitted.

The orientation of the contact normals in the two-dimensional planar space revealed itself as the most informative feature in our experiments and is used as input feature in the current implementation of our model. Note, that the normal of the object surface does not necessarily coincide with the normal of the sensor surface, due to the rigid fingers.

Positional information of the object is not utilized by the model yet, as only rotational estimates and corrections need to be performed. These positional information will be important for translational estimates, subject of future research.

Curvature is modeled by the eigenvalue ratio of the covariance of the tactile pressure "blobs", along with the angle of the first eigenvector. The information about the curvature of the contact area is very noisy and suffers from tangential stress (i.e. shear forces) along the sensor-surface which lead to strong distortions of the perceived pressure patterns. Thus, information extracted from the covariance of the pressure blobs did not lead to any improvements in the results, but may by used in the model with a different hardware, in which curvature measurement is less effected by the tangential stress.

D. The Problem of Mapping and Tracking

The extracted haptic features are transformed into the object coordinate system, based on the current estimate of the object pose. For mapping the object shape in a pose invariant representation, the tactile information is temporally integrated by the proposed neural-dynamic model, as shown in Fig. 1.

The central part of this architecture is the matching module for detecting and correcting errors in the orientation estimate according to the current tactile inputs and the object model. This requires to split the error into suitable proportions of shape model and pose model adaptation - which is the core problem to be addressed by SLAM.

This problem, when to adapt the object representation and when to correct the pose estimate, can only be solved to the degree of object ambiguity. The objects we used in our experiments, as well as many everyday objects, are symmetric and repetitive in their appearance, which makes localization of the object necessarily ambiguous, i.e. the same features may be detected for multiple object poses. The correction of the object pose estimate based on the tactile inputs corresponds to a tactile tracking of the object and is visualized by the "Tracking" path in Fig. 1.

A neural-dynamic implementation of the computations in the central modules of the architecture ("Object Representation" and "Matching Module" in Fig. 1) will be presented in the next Section.

III. THE MODEL

In this Section we describe the neural-dynamic model, which implements the object shape mapping and tracking

of its orientation, whereas Section IV describes the robotic experiments in which the model was evaluated.

A. Dynamic Neural Fields (DNFs)

In our architecture, DNFs are used to represent the current tactile features, to match these to an accumulated long-term memory of the object shape features, and to compute errors in the orientation estimation, thus stabilizing the object-centered shape representation.

Dynamic Neural Fields (DNFs) are activation functions, which first were introduced to describe activity of neuronal populations [1], [34] and have been used in cognitive science to model dynamics and development of cognitive processes, such as, e.g., memory formation, decision making, or categorization [30]. DNFs were first applied in a robotic context in the attractor dynamics approach to navigation, where they were used to stabilize target representation during occlusions [3]. Today, DNFs are one of the main tools in the dynamical systems approach to cognitive robotics and enable integration of low-level sensory inputs and motor dynamics into cognitive architectures, e.g., scene representation, sequence generation, and grounded language [28].

The DNFs used in our model follow Eq. (1), which defines the rate of change in activation $u(x, t)$ of the field:

$$\tau \dot{u}(x, t) = -u(x, t) + h + S(x, t) + \int f\big(u(x', t)\big)\omega(|x - x'|)dx'. \tag{1}$$

In Eq. (1), $u(x, t)$ is the activation of the DNF at time step t and position x. The position in a DNF describes a behavioral variable, such as a perceptual feature, location in space, or motor control variable (it is the orientation of the detected feature in our implementation here). The activation $u(x, t)$ can be interpreted as the confidence of value x for this behavioral variable in the current state.

The term $-u(x, t)$ stabilizes an attractor for the activation function at values, defined by the last three terms in the equation. The time constant τ determines how fast activation $u(x, t)$ relaxes to the attractor. The negative resting level h ensures that the DNF produces no output in a deactivated state and $S(x, t)$ is an external input, driving the DNF. The convolution term models lateral interactions between sites of an active DNF, shaped by the interaction kernel, $\omega(|x - x'|) = c_{exc} \exp\left[-\frac{(x - x')^2}{2\sigma_{exc}^2}\right] - c_{inh} \exp\left[-\frac{(x - x')^2}{2\sigma_{inh}^2}\right]$, with a short-range excitation (strength c_{exc}, width σ_{exc}) and a long-range inhibition (strength c_{inh}, width σ_{inh}). A sigmoidal non-linearity, $f\big(u(x, t)\big) = \frac{1}{1+\exp[-\beta u(x,t)]}$ defines the output of the DNF with which the DNF impacts on other dynamics in the model, as well as on its own dynamics through the lateral interactions.

The lateral interactions of DNFs stabilize a localized peak-attractor for the activation function, i.e. even for a noisy and varying input, the DNF "stabilizes a decision" for the most active peak position, leading to discretization of continuous sensory and motor spaces.

In the proposed model, a series of one-dimensional DNFs are used, encoding the orientation angle in the behavioral variable x in Eq. 1. A two-dimensional DNF is used to estimate the deviation of an activity peak position in two fields. This comparison mechanism has been previously used in a neurally-inspired model for coordinate frame transformations [29]. Hereby, both one-dimensional fields are expanded (i.e. projected) along a second dimension, one of them is transposed (i.e. rotated by $90°$) and then they are added up. This two-dimensional combined field activation serves as input for a two-dimensional field, defined by Eq.1 with x now $\vec{x} \in \mathbb{R}^2$. Finally, the two-dimensional field is projected along its diagonal onto a one-dimensional field, hence resulting in an encoding of one of the two fields relative to the other.

To build a long-term memory of the object's shape, we use memory trace dynamics, Eq. (2), [27]:

$$\tau_l \dot{P}(x, t) = \lambda_{build}\big(-P(x, t) + f(u(x, t))\big)f(u(x, t)) - \lambda_{decay}P(x, t)\big(1 - f(u(x, t))\big). \tag{2}$$

Here, $P(x, t)$ is the strength of the memory trace at site x of the DNF with activity $u(x, t)$ and output $f\big(u(x, t)\big)$, λ_{build} and λ_{decay} are the rates of build-up and decay of the memory trace. The build-up of the memory trace is active on sites with a high positive output $f\big(u(x, t)\big)$, the decay is active on the sites with a low output.

B. Mapping and Tracking the Object

The distribution of the tactile features (in particular, the normals' orientations, see Section II) within a short time window is given as input to two pathways of the model, implementing the mapping and tracking of the object. For this, a filter bank of orientation selective neurons implements a population encoding via a scalar product of the fixed neuron weight vectors and the contact normal. This encoding is then used as input for the subsequent DNFs. The first pathway is fast (i.e. has a small τ) and holds the current information of perceived tactile features, while the second pathway is operating at a slower speed and accumulates a long-term memory of past inputs. The latter serves as the object shape representation (i.e. mapping) and is used for matching with the current estimate of the fast pathway, in order to correct for errors in the orientation estimate (i.e. tracking). The neural-dynamic model, which implements both pathways is shown in Fig. 2.

a) Surface Detection: The fist step in both pathways is to classify, whether the current input comes from a flat surface by generating an activation peak at the location of the DNF, which corresponds to the orientation of the detected surface. As the finger surface is curved, flat object surfaces lead to more normals with the same orientation, while object edges lead to a continuous change in the orientation of the normals. Therefore, the distribution of contact normal orientations is broad and weak for contacts with an object edge, while it is sharp and strong for contacts with flat surfaces. The dynamics of the neural fields can easily be tuned to only give rise to a stable peak when a
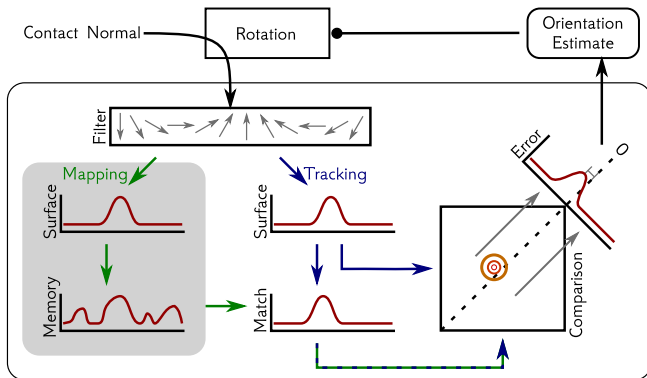
Fig. 2. Overview of the Model - An orientation selective filter bank is input for a mapping and a tracking pathway, consisting of multiple DNFs. Mapping in the gray underlaid DNF and MT corresponds to the object model and operates with a slower time constant. Tracking utilizes the memory and outputs a correction term for the estimated object orientation.

Fig. 3. Left: sketch of the rotation behaviour; Right: picture of our experimental setup

sufficient number of measurements with the same orientation is reached.

b) Memory and Matching: In the mapping pathway, the activation peak, which represents a detected surface, is transferred into memory (Fig. 2), through the memory trace dynamics specified in Eq. (2). The memory is exponentially fading and is tuned to have very slow fading time constant in order to hold sufficient information of past orientations of the detected surfaces.

In the tracking pathway, the orientation of a detected surface is transferred to a match DNF (Fig. 2). The lateral interactions in this DNF shift the representation of the currently perceived surface towards the position of a neighboring peak in the memory. This happens if the activation peak of the currently detected surface has a similar position (i.e. orientation) in the match DNF as a previously detected surface, stored in the memory.

c) Error Estimation and Correction: The matched activation peak represents the corrected orientation and is compared to the original activation peak in a two-dimensional DNF (labeled "Comparison" in Fig. 2) in order to determine if the matching process altered the peak position, as described in the previous paragraph. A diagonal readout of the two-dimensional DNF, i.e. projection to a one-dimensional Error DNF (Fig. 2), provides information of the peak shift due to the matching.

The deviation of the peak position from the center of the Error DNF is subsequently used to correct the current estimate of the object's orientation (encoded in a scalar value). Subsequently, the corrected estimate of the orientation is then used to transform the next perceived feature into the corrected object-centered reference frame.

Together, the two pathways lead to a fast matching of current with memorized features and a correction in the orientation estimate.

IV. EXPERIMENTS

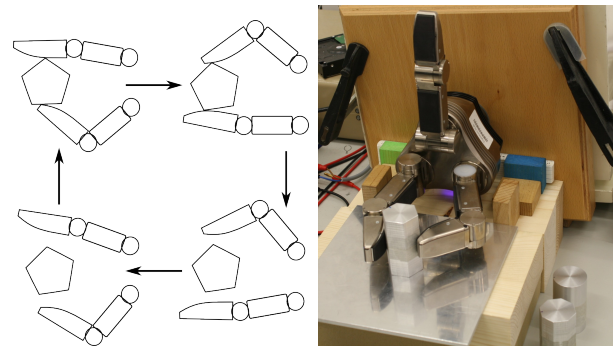In our experimental setup a Shunk Dexterous Hand 2 (SDH-2) is used and configured such that only two of the three fingers are used, each having two degrees of freedom (i.e. controlled joints). The two phalanges of the fingers are each equipped with a tactile sensor. The tactile sensors, used in our experiments, consist of an array of 6×13 tactile elements (texels) on the distal phalanges, although the width decreases to 4 texels at the fingertips. Figure 3 shows the robotic setup, used for evaluation of the model, as well as the manipulation behavior used in our experiments.

Rotation experiments were performed with three different aluminum objects (n-gons): a round cylinder, an 8-sided and a 6-sided cylinder which had the same medium diameter (4.0cm) and object height (7.0cm). The objects had a hole in the bottom, by which they were attached to a steel axis to prevent translations of the object. This leaves one degree of freedom to study errors accumulated during the object manipulation: a rotation along the fixed axis.

A. Generated Datasets

With each of the objects, five datasets were recorded, each consisting of an estimated rotation of four times 360 degrees. Hence, 15 datasets were collected, in which the tactile patterns and joint angles were sampled with approximately 2-3 Hz and the according features were computed and stored.

An exemplary subset of two rotations is visualized in Fig. 4 from the first dataset of the six-sided object, where finger one corresponds to the upper finger of the sketch in Fig. 3. Shown are the positions and surface normals of the tactile contacts during two rotations of the object for each finger, respectively.

It is clearly visible, that there is a drift in the object rotation estimate as the data points do not align for consecutive full rotations of the object. The disparity in the noise level of computed contact curvatures of the fingers (not shown) indicated that noise is affected by the direction of tangential forces, induced by the movement direction of the fingers.

Using these noisy measurements directly to build an object's shape representation would lead to large errors in this representation accumulated over time, which motivates application of our neurally-inspired model to correct the measurements in an online learning process.
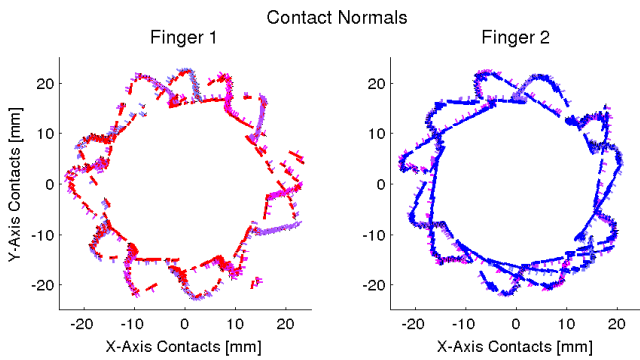
Fig. 4. Raw, uncorrected data for a six-sided object. The position and orientation of the contact normals in the object coordinate system (used as input to the model). The columns show the data for finger one and two, respectively. Note how errors in position estimates lead to a drift of features over time of manipulation (two full rotations of the object are shown here).

V. RESULTS

To evaluate the benefit of the neural-dynamic model presented in this paper, three different estimators of object shape, i.e. the number of detected surfaces during manipulation, were used, each operating on the tactile input to the model:

First, the accumulated histogram of all past contact normal orientations is computed. This is the most simple approach to classify the number of surfaces based on orientations, which does not use the memory trace and the error correction mechanism of our model.

Second, the memory trace of the proposed model is used for evaluation, however without any correction in the pose estimates. In this case, the memory performs a suppression of weak (i.e. small) surfaces and additionally implements a fading memory.

Third, the memory trace of the model with error correction. Here the full model of Fig. 2 is used and continuously matches the current orientation of a detected surface with the memory and outputs a correction term for the pose estimation.

In Fig. 5, the time-courses of the histogram, the uncorrected memory trace, and the corrected memory trace are shown as the rotation behavior is performed for the six-sided object. In the histogram approach, each row (i.e. every
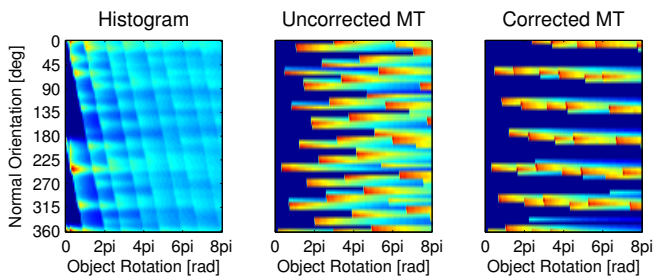

Fig. 5. The histogram, uncorrected memory trace and corrected memory trace for dataset 5 of the six-sided object. Note that the y-axis is circular and should have six equally spaced peaks aligned for all rotation steps.
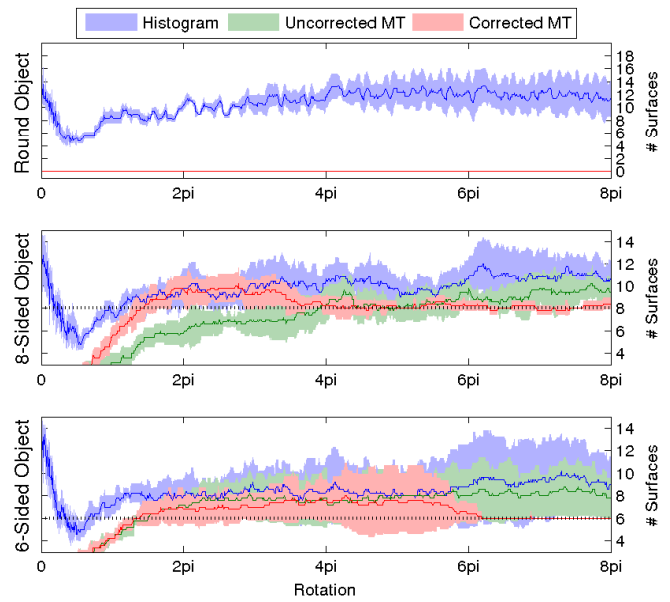

Fig. 6. The mean and standard deviation of the number of estimated object surfaces, evolving over time (i.e. object rotations). For each of the three objects (subplots) five datasets were used for computing the graphs with three different methods, respectively. Blue: based on an accumulated histogram. Green: based on the proposed model with deactivated error feedback. Red: based on the proposed model with error correction.

rotation step) is normalized for an increased visibility. The memory is intrinsically normalized, as given by Eq. 2. Note the clear increase in alignment of the detected surfaces during the rotation when the model performs corrections in the pose estimate ("Corrected MT" in Fig. 5).

Each estimator was analyzed in order to determine the number of detected surfaces for each dataset. First, the activation of the estimator was smoothed with a Gaussian filter in order to remove local optima due to small amounts of noise. Second, the number of peaks above a predefined threshold (0.4) are counted using Matlab Signal Processing Toolbox. The results are not sensitive to the exact threshold value, as peaks, which correspond to surfaces typically have values from 0.6 to 0.8.

Figure 6 shows the mean and standard deviation of the number of estimated surfaces using the three estimators. The number of surfaces detected is shown for each measurement step (object manipulation action) and is computed by averaging the five datasets for each object.

In general, the simple accumulated histogram (first estimator) and the uncorrected model (second estimator) are incapable of building a consistent representation of the object shape, as the errors in the pose estimate are integrated and lead to a constant drift and the number of detected surfaces strongly fluctuates over time (see Fig. 6, blue and green lines). Note, how only memory trace with error correction (full model) converges to a correct estimate of the number of surfaces of the objects (red line on the three plots in Fig. 6).

As the Memory is empty in the beginning and only incrementally builds up, the number of surfaces is typically
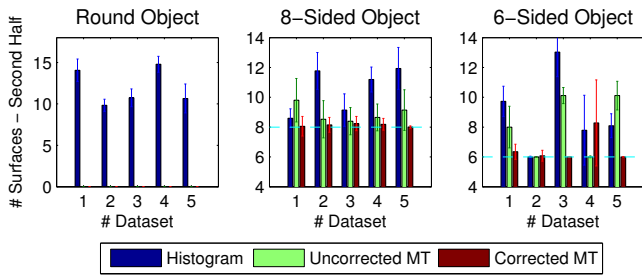
Fig. 7. The mean and standard deviation of the number of detected surfaces for the second half of each dataset, excluding the initial "build up phase".
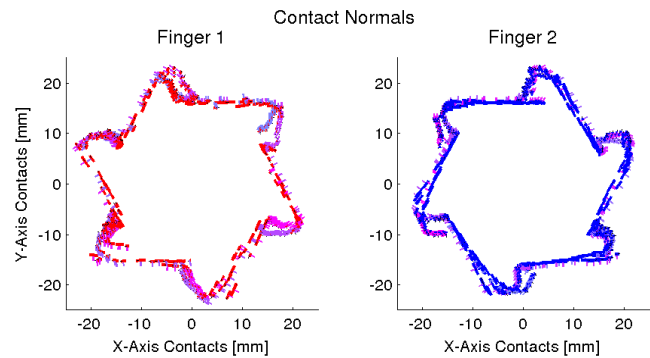

Fig. 8. The corrected feature measurements for a six-sided object. The correction of the estimated location of features in the object-centered reference frame by the neural-dynamic model for haptic learning aligns feature measurements over several (here, two are shown) full rotations of the object. See Fig. 4 for a comparison.

overestimated after the first 360 degrees of the rotation (red curve in Fig. 6). This is because every new surface is "corrected" into the direction of the previously seen surface and has no counterpart on the other side jet. However, during further exploration of the object the model shifts and merges the orientations of surfaces in the Memory and finally converges to a stable representation, as can be seen in Fig. 6, red lines. Note, that this effect is not visible in the dataset shown in Fig 5.

For the round object, the contact normals only lead to a low activation in the surface detection DNF which in turn leads to a memory formation which is sub-threshold. Therefore, no surfaces are found by the proposed model in any of the datasets with the round object, which is a clear advantage over the histogram approach, which detects several false surfaces.

Figure 7 shows the mean and standard deviation of the detected number of surfaces for the second half of each dataset of the recorded data. The first half is omitted in order to give the model time for building up the representation. The corrected memory trace model shows improved performance compared to other approaches, see red bars in Fig. 7. The forth dataset of the six-sided object shows a worse performance, due to the late convergence of the object shape after approximately three full rotations (around 6pi in Fig. 6). Nevertheless, even in this case, the correct number of surfaces is detected and sustained by the full model after three full rotations (Fig. 6).

Corresponding to finding the right number of surfaces, the model also finds an appropriate correction for the orientation estimate of the object. Fig. 8 shows an improvement in alignment of measurements, accumulated over time compared to the raw data in Fig. 4. The shown data are the last two full rotations of dataset one of the six-sided object, with the corrected orientation estimate incorporated. Compared to the uncorrected raw data (Fig. 4), the surfaces detected during two full rotations are aligned much more precisely.

VI. DISCUSSION

Learning an object representation and tracking its pose is crucial for improving and planing manipulations, as it enables the prediction of the outcome of movements. In this paper, we proposed an online, neurally inspired model capable of learning an object model from purely haptic data.

The object model amounts to a representation of features, relevant for object manipulation in an object-centered space. Evaluation showed that the full model clearly increases the object shape precision, compared to directly integrating the raw data.

Here, we discuss several limitations of the current implementation and the steps we planned to overcome those.

Going towards three-dimensional haptic SLAM, i.e. localizing the rotation and position in two-dimensional space, is a necessary step for enabling the prediction of tactile sensor inputs from efferent motor signals. Thus, one of the next steps in our work is to incorporate information about translational movement into the model. Currently, the information of the position in space of tactile contacts was not used and the object position was neither estimated nor corrected by the model. However, the position of tactile contacts is available along with the orientation of the contacts, which was used in this paper, and may be incorporated in order to independently estimate the object position.

It is obvious that the sensitivity of the matching process determines the spatial resolution of the features which can be detected. Features in the present model are only distinguishable by their pose, which is exactly the variable in need of correction. The matching process in the Match DNF pulls an activity peak, which represents a surface, towards the neighboring peak in the Memory DNF. If the matching neighborhood is too broad, distinct features stemming from different surfaces, may be matched and the pose estimate is "corrected" in order to align those. On the other hand, if the matching neighborhood is too narrow, revisiting the same feature once again will lead to a new feature in the Memory, due to perturbations in the true object position.

In order to overcome these limitations, higher order features, which may consist of specific combinations of lower level features, should be considered. More specifically, combinations of surfaces and edges as well as the angle of an edge (distance between the surface-peaks in the histogram) will be investigated in future research. Additionally, improv-

ing the feature of the contact surface curvature by removing "ghost contacts" is expected to further improve the results, due to a richer representation of tactile inputs. We have used this feature already in test runs, but it revealed itself to be too noisy with the currently used hardware.

The final goal of our project is to close the loop and let the accumulated representation of the object's shape influence the manipulation behavior, making grasps which follow a given pattern, e.g. falling on flat surfaces only. To accomplish this, the acquired object's shape representation may be used to generate predictions of the outcome of the robot's actions.

The present work is a first, necessary step into the direction of autonomous haptic learning of objects' shapes. We hope that future work will further increase our understanding of tactile features and memory processes necessary to tackle the main challenges in haptic SLAM, in artificial as well as biological systems.

References

[1] S Amari. Dynamics of pattern formation in lateral-inhibition type neural fields. *Biological Cybernetics*, 27:77–87, 1977.

[2] Tim Bailey and Hugh Durrant-Whyte. Simultaneous localization and mapping (slam): Part ii. *Robotics & Automation Magazine, IEEE*, 13(3):108–117, 2006.

[3] E Bicho, P Mallet, and G Schöner. Using attractor dynamics to control autonomous vehicle motion. In *Proceedings of IECON'98*, pages 1176–1181. IEEE Industrial Electronics Society, 1998.

[4] Anna Bodegård, Stefan Geyer, Christian Grefkes, Karl Zilles, and Per E Roland. Hierarchical processing of tactile shape in the human brain. *Neuron*, 31(2):317–328, 2001.

[5] Maxime Chalon, Jens Reinecke, and Martin Pfanne. Online in-hand object localization. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 2977–2984. IEEE, 2013.

[6] Nicolas Cuperlier, Mathias Quoy, and Philippe Gaussier. Neurobiologically inspired mobile robot navigation and planning. *Frontiers in neurorobotics*, 1, 2007.

[7] Stanimir Dragiev, Marc Toussaint, and Michael Gienger. Gaussian process implicit surfaces for shape estimation and grasping. In *Robotics and Automation (ICRA), International Conference on*, pages 2845–2850. IEEE, 2011.

[8] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *Robotics & Automation Magazine, IEEE*, 13(2):99–110, 2006.

[9] Charles Fox, Mat Evans, Martin Pearson, and Tony Prescott. Tactile slam with a biomimetic whiskered robot. In *Robotics and Automation (ICRA), International Conference on*, pages 4925–4930. IEEE, 2012.

[10] Paul Hebert, Nicolas Hudson, Jeremy Ma, and Joel Burdick. Fusion of stereo vision, force-torque, and joint sensors for estimation of in-hand object location. In *Robotics and Automation (ICRA), International Conference on*, pages 5935–5941. IEEE, 2011.

[11] Steven Hsiao. Central mechanisms of tactile shape perception. *Current opinion in neurobiology*, 18(4):418, 2008.

[12] Steven Hsiao and M. Gomez-Ramirez. *Neurobiology of Sensation and Reward - Chap. 7: Touch*. Frontiers in Neuroscience. CRC Press, 2011.

[13] Chia-Chun Hung, Eric T Carlson, and Charles E Connor. Medial axis shape coding in macaque inferotemporal cortex. *Neuron*, 74(6):1099–1113, 2012.

[14] Thomas W James, Sunah Kim, and Jerry S Fisher. The neural basis of haptic object processing. *Canadian Journal of Experimental Psychology*, 61(3):219–229, 2007.

[15] Roland S Johansson and J Randall Flanagan. Coding and use of tactile signals from the fingertips in object manipulation tasks. *Nature Reviews Neuroscience*, 10(5):345–359, 2009.

[16] Michael C Koval, Mehmet R Dogar, Nancy Pollard, and Siddhartha Srinivasa. Pose estimation for contact manipulation with manifold particle filters. *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, 2013.

[17] N Krüger, P Janssen, S Kalkan, M Lappe, A Leonardis, J Piater, AJ Rodriguez-Sanchez, and L Wiskott. Deep hierarchies in the primate visual cortex: What can we learn for computer vision? *IEEE transactions on pattern analysis and machine intelligence*, 2012.

[18] Qiang Li, Carsten Schürmann, Robert Haschke, and Helge Ritter. A control framework for tactile servoing. In *Proc. Robotics: Science and Systems*, 2013.

[19] Yangming Li, Shuai Li, and Yunjian Ge. A biologically inspired solution to simultaneous localization and consistent mapping in dynamic environments. *Neurocomputing*, 2012.

[20] Martin Meier, Matthias Schopfer, Robert Haschke, and Helge Ritter. A probabilistic approach to tactile shape reconstruction. *Robotics, IEEE Transactions on*, 27(3):630–635, 2011.

[21] Michael J Milford, Gordon F Wyeth, and David Prasser. Ratslam: a hippocampal model for simultaneous localization and mapping. In *Robotics and Automation, ICRA, International Conference on*, pages 403–408. IEEE, 2004.

[22] A Miquée, C Xerri, C Rainville, J-L Anton, B Nazarian, M Roth, and Y Zennou-Azogui. Neuronal substrates of haptic shape encoding and matching: a functional magnetic resonance imaging study. *Neuroscience*, 152(1):29–39, 2008.

[23] Zachary Pezzementi, Caitlin Reyda, and Gregory D Hager. Object mapping, recognition, and localization from tactile geometry. In *Robotics and Automation (ICRA), International Conference on*, pages 5942–5948. IEEE, 2011.

[24] Robert Platt, Leslie Kaelbling, Tomas Lozano-Perez, and Russ Tedrake. Simultaneous localization and grasping using belief space planning. In *Workshop on Manipulation Under Uncertainty, Robotics and Automation (ICRA), International Conference on*, 2011.

[25] Mila Popović, Dirk Kraft, Leon Bodenhagen, Emre Başeski, Nicolas Pugeault, Danica Kragic, Tamim Asfour, and Norbert Krüger. A strategy for grasping unknown objects based on co-planarity and colour information. *Robotics and Autonomous Systems*, 58(5):551–565, 2010.

[26] M. Richter, Y. Sandamirskaya, and G Schöner. A robotic architecture for action selection and behavioral organization inspired by human cognition. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2012.

[27] Yulia Sandamirskaya. Dynamic neural fields as a step towards cognitive neuromorphic architectures. *Frontiers in Neuroscience*, 7:276, 2013.

[28] Yulia Sandamirskaya, Stephan K.U. Zibner, Sebastian Schneegans, and Gregor Schöner. Using dynamic field theory to extend the embodiment stance toward higher cognition. *New Ideas in Psychology*, 31(3):322 – 339, 2013.

[29] Sebastian Schneegans and Gregor Schöner. A neural mechanism for coordinate transformation predicts pre-saccadic remapping. *Biological cybernetics*, 106(2):89–109, February 2012.

[30] G Schöner. Dynamical Systems Approaches to Cognition. In Ron Sun, editor, *Cambridge Handbook of Computational Cognitive Modeling*, pages 101–126, Cambridge, UK, 2008. Cambridge University Press.

[31] Linda B Smith. From fragments to geometric shape changes in visual object recognition between 18 and 24 months. *Current Directions in Psychological Science*, 18(5):290–294, 2009.

[32] Jacqueline C Snow, Lars Strother, and Glyn W Humphreys. Haptic shape processing in visual cortex. *Journal of Cognitive Neuroscience*, 2013.

[33] Elizabeth Spelke, Sang Ah Lee, and Véronique Izard. Beyond core knowledge: Natural geometry. *Cognitive Science*, 34(5):863–884, 2010.

[34] H R Wilson and J D Cowan. A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue. *Kybernetik*, 13:55–80, 1973.

[35] Li Zhang, Siwei Lyu, and Jeff Trinkle. A dynamic bayesian approach to real-time estimation and filtering in grasp acquisition. In *Robotics and Automation (ICRA), International Conference on*, pages 85–92. IEEE, 2013.