Active Estimation of Object Dynamics Parameters with Tactile Sensors

Hannes P. Saal, Jo-Anne Ting, and Sethu Vijayakumar

***Abstract*— The estimation of parameters that affect the dynamics of objects—such as viscosity or internal degrees of freedom—is an important step in autonomous and dexterous robotic manipulation of objects. However, accurate and efficient estimation of these object parameters may be challenging due to complex, highly nonlinear underlying physical processes. To improve on the quality of otherwise hand-crafted solutions, automatic generation of control strategies can be helpful. We present a framework that uses active learning to help with sequential gathering of data samples, using information-theoretic criteria to find the optimal actions to perform at each time step. We demonstrate the usefulness of our approach on a robotic hand-arm setup, where the task involves shaking bottles of different liquids in order to determine the liquid's viscosity from only tactile feedback. We optimize the shaking frequency and the rotation angle of shaking in an online manner in order to speed up convergence of estimates.**

I. INTRODUCTION

In application domains such as service robotics, haptics, medical and surgical robotics, and human-robot interaction, dextrous manipulation of objects is crucial. In the absence of visual information, force and tactile sensory data are typically the only feedback available. Since not all the dynamic properties of objects will be known beforehand, the robot may have to estimate some of the object parameters (affecting object dynamics). For example, when handling open containers filled with fluids, the type of liquid constrains the kinds of movement a robot can make without any spillage. Quick and reliable estimation of such object parameters is, thus, important for bringing robots into real-world environments.

To achieve this, the robot needs to be equipped with a sensor model that maps both the object parameter of interest and the robot's current state to sensor readings. Assuming the sensor model is known (either learned from data or modeled with physics-based models), the robot is faced with the challenge of determining a sequence of actions to perform in order to gather maximally informative observations. The smallest set of informative observations is desired so that the object parameter of interest can be estimated as quickly and accurately as possible. We are interested in application domains where observations and actions are potentially high-dimensional and continuous and where decisions need to be made quickly, often in real-time.

In this paper, we use active learning in order to speed up the sequential estimation of the viscosity of various liquids.

H.P.S. and S.V. are with the School of Informatics, University of Edinburgh, Edinburgh EH8 9AB, UK; J.T. is with the Department of Computer Science, University of British Columbia, Vancouver, BC, V6T 1Z4, Canada. Emails: hannes.saal@ed.ac.uk, jting@acm.org and svijayak@inf.ed.ac.uk

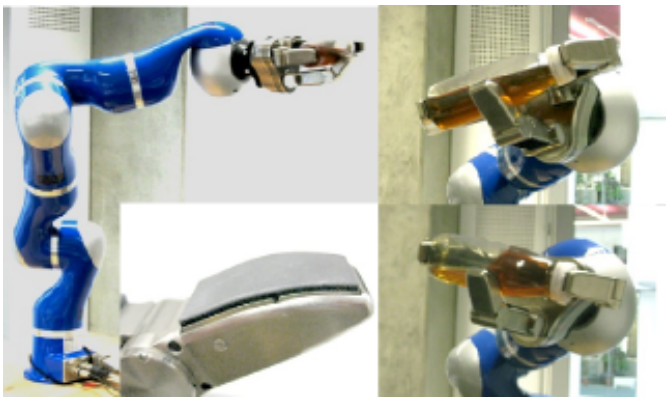


Fig. 1. Left: Schunk SDH 7-DOF hand mounted on DLR 7-DOF robotic arm. Inset: Touch sensor array on robotic finger. Right: Liquid-filled bottles are shaken at different frequencies and rotation angles. Top right: Positive rotation angle, with bottle tilted upwards. Bottom right: Negative angle, leading to downwards tilt.

Filled bottles are gripped and then shaken by a robotic hand-arm system, shown in Figure 1, while tactile feedback from touch sensor arrays mounted on the robotic fingers is used to determine the viscosity of the liquids. Crucially, we optimize the robot's actions (e.g., shaking frequencies and rotation angle, as shown in Figure 1) at each time step in order to maximize the informativeness of the next incoming observation.

In our proposed framework, we learn the sensor model from training data, using sparse Gaussian Processes (GPs) [1] to achieve efficient belief updating and analytical calculations of information gains. We use an information-theoretic criterion to calculate the optimal action to perform at each time step, and our algorithm can perform this computation quickly due to the use of sparse GPs. The basic theoretical framework and results on a simpler robotic experiment have been presented in [2]. Here, we extend these methods by incorporating sparse GPs, allowing us to include thousands of training data points and thereby extend the dimensionality of the action space. This allows us to tackle problems where brute-force optimization of the mutual information becomes untenable. Moreover, we provide details on the robotic implementation and discuss strategies for viscosity estimation.

In the following section, we give an overview of related approaches. We describe our proposed framework in section III before presenting experimental results in section IV. Evaluations are shown for a liquid viscosity estimation task using the DLR-Schunk robotic hand-arm system.

II. RELATED APPROACHES IN ROBOTICS

We first review recent approaches to estimation of dynamic object properties and then provide a brief overview of past work on sequential active learning in robotics.

A. Dynamic estimation of object properties

Early studies have looked into estimating the weight, center of mass, and modes of inertia to loads attached to a robot arm [3]. These quantities can be estimated from rigid body dynamics equations, and in these studies, observations were collected by applying specific, parameterized trajectories chosen by the experimenters. Object properties like hardness and texture—which are difficult or impossible to determine from static grasps—have also been estimated with dynamic movements [4]. Some studies used shaking movements in order to discriminate between objects, but movement parameter values (like shaking frequency) were hand-tuned by experimenters [5], [6]. An active strategy to select the movement parameters would clearly be beneficial in this case. While some of the studies mentioned above deal with estimation of continuous-valued object parameters, most focus on the task of discrimination within a discrete set of objects.

B. Active sequential learning in robotics

Active learning has been applied to the problem of setting camera parameters such as zoom and pan when discriminating between different objects [7]. The solution relied on either discretization or Monte Carlo sampling for computing information gain, making it hard to scale to higher-dimensional problems found in real-world domains. Another use of active learning involves generating internal models of the robot's current configuration (e.g. for detecting missing limbs) by performing discriminatory actions [8].

Past work has also used reinforcement learning [9] or evolutionary algorithms [10] in order to learn control policies that allow for quick discrimination of objects or parameter discrimination. These methods differ from our approach since we determine optimal actions to be taken in an online manner (i.e., during run-time). While this means that the optimal action to be performed needs to be computed at each time step, our approach makes it easy to add more training data (i.e., to gather more observations) without having to re-learn the policy (i.e., the sequence of actions to be taken). In contrast to past work that optimize actions and perceptions together, we de-couple the optimization (learning the sensor model and optimal actions in separate phases) so that discrimination can be done even when sub-optimal actions are taken.

Gaussian processes [11] have been used in active learning before, but most applications address how a sparse(r) set of training data can be selected from a much larger data set while retaining as much accuracy in the regression model as possible, e.g., [12]. Another notable use includes the use of GPs to model the cost function, which is then used to trade of exploration and exploitation [13].

III. METHODS

A. Problem formulation

We assume the following notation:

- $\boldsymbol{\theta} \in \mathbb{R}^{d_{\boldsymbol{\theta}}}$ is the parameter of interest that we want to estimate (here, viscosity of a liquid).
- $\mathbf{x} \in \mathbb{R}^{d_{\mathbf{x}}}$ is a vector of action parameters (shaking frequency and rotation angle).
- $\mathbf{y} \in \mathbb{R}^{d_{\mathbf{y}}}$ is observed sensory data.

We assume that observations $\mathbf{y}$ are a (nonlinear) function of both the actions $\mathbf{x}$ and state parameters $\boldsymbol{\theta}$:

$$\mathbf{y} = f\left(\mathbf{x}, \boldsymbol{\theta}\right) + \epsilon_{\mathbf{y}} \tag{1}$$

where $\{\mathbf{x}, \boldsymbol{\theta}, \mathbf{y}\}$ are all continuous, and $\epsilon_{\mathbf{y}}$ is observation noise. Both $\mathbf{x}$ and $\mathbf{y}$ are potentially high-dimensional.

For active learning, we are interested in determining the optimal actions $\mathbf{x}^*$ to take during test time such that the mutual information between $\mathbf{y}$ and $\boldsymbol{\theta}$, $\mathbf{I}\left(\boldsymbol{\theta}; \mathbf{y}|\mathbf{x}\right)$, is maximized, i.e., $\mathbf{x}^* = \arg\max_{\mathbf{x}\in\mathbf{X}} \mathbf{I}\left(\boldsymbol{\theta}; \mathbf{y}|\mathbf{x}\right)$. The mutual information is defined as:

$$\mathbf{I}\left(\boldsymbol{\theta}; \mathbf{y}|\mathbf{x}\right) = \iint p(\boldsymbol{\theta}, \mathbf{y}|\mathbf{x}) \log \frac{p(\boldsymbol{\theta}, \mathbf{y}|\mathbf{x})}{p(\boldsymbol{\theta}) p(\mathbf{y}|\mathbf{x})} d\mathbf{y} d\boldsymbol{\theta} \tag{2}$$

where $p(\boldsymbol{\theta}, \mathbf{y}|\mathbf{x})$ is the joint probability distribution of $\boldsymbol{\theta}$ and $\mathbf{y}|\mathbf{x}$; and $p(\boldsymbol{\theta})$ and $p(\mathbf{y}|\mathbf{x})$ are the marginal probability distributions of $\boldsymbol{\theta}$ and $\mathbf{y}$, respectively.

B. Learning the Sensor Model

To learn the sensor model, we use a GP to approximate the nonlinear function f in Eq. (1). GPs rely on a kernel function to determine the correlations between different input points. A kernel matrix $\boldsymbol{\Lambda}$ is computed from this kernel function, specifying the correlations between all training points. We learn a GP for each output dimension $m = 1, .., d_{\mathbf{y}}$ and use a squared exponential kernel function of the following form:

$$k_m\left(\mathbf{z}_p, \mathbf{z}_q\right) = \alpha_m^2 \exp\left\{\frac{1}{2}(\mathbf{z}_p - \mathbf{z}_q)^T \mathbf{H}_m^{-1} (\mathbf{z}_p - \mathbf{z}_q)\right\} + \sigma_m^2 \delta(\mathbf{z}_p, \mathbf{z}_q) \tag{3}$$

where $\mathbf{z}_p$ and $\mathbf{z}_q$ are the vectors of the form $[\boldsymbol{\theta}\ \mathbf{x}]^T$; $\mathbf{H}_m = \begin{pmatrix} \mathbf{H}_m^{\boldsymbol{\theta}} & 0 \\ 0 & \mathbf{H}_m^{\mathbf{x}} \end{pmatrix}$, $\mathbf{H}_m^{\boldsymbol{\theta}}$ and $\mathbf{H}_m^{\mathbf{x}}$ are diagonal matrices; α_m^2 is a scaling parameter; and σ_m^2 denotes the variance of additive noise. The set of hyperparameters Γ_m to be optimized for kernel function k_m is $\Gamma_m = \left\{\alpha_m^2, \sigma_m^2, \mathbf{H}_m^{\boldsymbol{\theta}}, \mathbf{H}_m^{\mathbf{x}}\right\}$.

When GPs are used for filtering [14], as in our approach, the resulting calculations have a computational complexity that is quadratic in the number of training points at each time step. As a result, updates can be slow if the number of training points is high. Sparse extensions of GPs rely on a small number of pseudo-inputs to speed up computations. We use one of these methods [1], where the location of pseudo-inputs are optimized and some effects of heteroscedastic noise are modeled.

Given the model specified by Eqs. (1) to (3), we can compute the mapping from given values of viscosity θ and action parameters $\mathbf{x}$ to a probability distribution over sensor

readings $\mathbf{y}$. In particular, the predictive distribution over $\mathbf{y}$ is Gaussian with predictive mean $\mathbf{m}_m^*$ and predictive variance Σ_m^*:

$$\mathbf{m}_m^* = \mathbf{k}_m^T \mathbf{W}_m^{-1} \mathbf{K}_m^{\uparrow} \mathbf{\Lambda}_m^{-1} \mathbf{y}_m \tag{4}$$

$$\Sigma_m^* = \sigma^2 - \mathbf{k}_m^T (\mathbf{K}_m^{-1} - \mathbf{W}_m^{-1}) \mathbf{k}_m, \tag{5}$$

where $\mathbf{W}_m = \mathbf{K}_m + \mathbf{K}_m^{\uparrow} \mathbf{\Lambda}_m^{-1} \mathbf{K}_m^{\downarrow}$. Here, $\mathbf{\Lambda}_m$ is the kernel matrix calculated from the training inputs, $\mathbf{K}_m$ is the kernel matrix for the pseudoinputs and $\mathbf{K}_m^{\uparrow}$ and $\mathbf{K}_m^{\downarrow}$ are $T \times U$ and $U \times T$ matrices, respectively, containing the correlations between pseudo-input and training inputs, where U is the number of pseudo-inputs and T is the number of training inputs.

The GPs are learned from training data by first optimizing the hyperparameters Γ_m of each kernel function k_m while keeping the location of the pseudo-inputs fixed, and then optimizing the position of the pseudo-inputs. Optimization is done by gradient descent on the model likelihood [1]. To prevent overfitting, we leave out some of the initial training data gathered to use as a validation set. Once the estimated error on this validation set starts to increase, we stop optimization.

C. Sequential Estimation

For detailed derivation, please refer to [2]. In this paper, we extend the approach there to sparse GPs.

The algorithm starts with a broad Gaussian initial prior over the viscosity $\boldsymbol{\theta}$ in order to reflect our ignorance about the viscosity of the liquid in the bottle. Optimal action parameters are then found that maximize the mutual information in Eq. (2), and the resulting observations are recorded under these new actions. Finally, the posterior distribution over the viscosity $\boldsymbol{\theta}$ is computed, leading to updated mean and covariance terms. The process is repeated iteratively, with the optimal actions being calculated at every time step. Updates to the viscosity's posterior mean and posterior covariance are similar to Kalman-filter like updates:

$$\boldsymbol{\mu}_{t+1} = \boldsymbol{\mu}_t + \mathbf{C}_{t+1}^T \mathbf{S}_{t+1}^{-1} (\mathbf{y}_{t+1}^{obs} - \mathbf{m}_{t+1}) \tag{6}$$

$$\mathbf{\Sigma}_{t+1} = \mathbf{\Sigma}_t - \mathbf{C}_{t+1}^T \mathbf{S}_{t+1}^{-1} \mathbf{C}_{t+1} \tag{7}$$

where $\boldsymbol{\mu}_t$ is the posterior mean of $\boldsymbol{\theta}$ at time step t and $\mathbf{\Sigma}_t$ is the corresponding posterior covariance. $\mathbf{y}^{obs}$ is the new sensory data. The terms $\mathbf{C}_{t+1}$, $\mathbf{S}_{t+1}$, and $\mathbf{m}_{t+1}$ correspond to the cross-covariance between $p(\boldsymbol{\theta}, \mathbf{x})$ and $p(\mathbf{y})$, and the marginal mean and variance of $p(\mathbf{y})$, respectively (see below).

We introduce an active component to the selection of action $\mathbf{x}_{t+1}$ at time step t by maximizing the mutual information between the current probability distribution over $\boldsymbol{\theta}$ and future observations, conditioned on the actions taken. The optimal action $\mathbf{x}_{t+1}^*$ is found as follows:

$$\begin{aligned} \mathbf{x}_{t+1}^* &= \text{argmax}_{\mathbf{x}_{t+1}} \ \mathbf{I}(\boldsymbol{\theta}_t; \mathbf{y}_{t+1} | \mathbf{x}_{t+1}) \\ &= \text{argmax}_{\mathbf{x}_{t+1}} \ |\mathbf{C}_{t+1}(\mathbf{x}) \mathbf{S}_{t+1}(\mathbf{x}) \mathbf{C}_{t+1}(\mathbf{x})^T| \end{aligned} \tag{8}$$

where the above is maximized by performing gradient ascent.

For the rest of the paper, we drop the time index (subscript t) on $\mathbf{C}$, $\mathbf{S}$, and $\mathbf{m}$. Since the sensor model is non-linear, these quantities have to be approximated, e.g., as in the unscented Kalman filter [15]. However, it has been shown that these quantities can be calculated analytically when approximating $p(\mathbf{y})$ with a Gaussian distribution [16], [14]:

$$m_m = \mathbf{q}_m(\mathbf{x})^T \mathbf{a}_m \tag{9}$$

$$\begin{aligned} S_{mn} = \mathbf{a}_m^T \mathbf{Q}_{mn}(\mathbf{x}) \mathbf{a}_m - m_m m_n \\ + \delta(m-n) \left(\alpha_m^2 - tr(\mathbf{B}_m \mathbf{Q}_{mm}) \right) \end{aligned} \tag{10}$$

$$C_{mn} = \mathbf{Z}_m^T(\mathbf{x}) \mathbf{a}_m - \mu_n m_m \tag{11}$$

where m_m is the m-th coefficient of $\mathbf{m}$; S_{mn} is the (m,n)-th entry of the matrix $\mathbf{S}$; C_{mn} is the (m,n)-th entry of the matrix $\mathbf{C}$. Moreover, $\mathbf{a}_m = \mathbf{W}_m^{-1} \mathbf{K}_m^{\uparrow} \mathbf{\Lambda}_m^{-1} \mathbf{y}_m$ and $\mathbf{B}_m = \mathbf{K}_m^{-1} - \mathbf{W}_m^{-1}$ (see section III-B). For definitions of $\mathbf{q}_m$, $\mathbf{Q}_{mn}$, and $\mathbf{Z}_m$, as well as derivatives, please refer to [2].

D. Implementation

During run-time, optimization of the actions has to be performed as quickly as possible. We are able to compute this quickly for following reasons:

- During gradient ascent, only the terms dependent on $\mathbf{x}$ are recalculated, thereby speeding up the calculations.
- The optimization can be run in parallel from different starting points.
- We place an upper limit on the run-time of the optimization at each time step (500 msec) and stop when this upper limit has been reached. (Please refer to section IV-B for why 500 msec was chosen.) Since we perform gradient ascent, this ensures that whenever we stop the optimization, we will have improved on the informativeness of the next action.
- We assumed hyperparameters are shared across all output dimensions, so that we can re-use kernel matrices and reduce computation time.

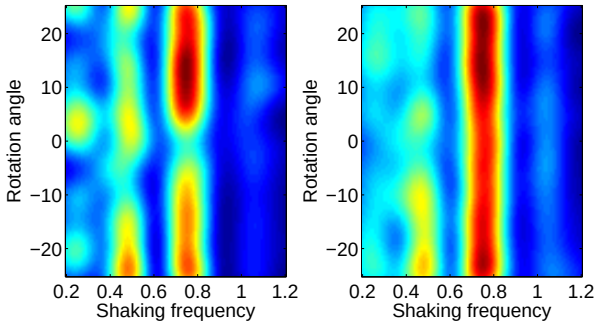
IV. EXPERIMENTAL RESULTS

A. Setup and preprocessing

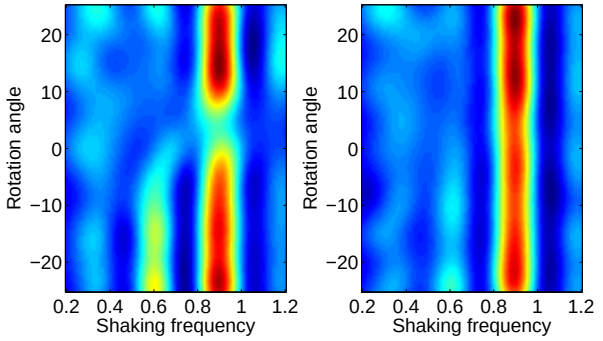
We examined the problem of viscosity estimation from tactile sensory data using the robotic anthropomorphic hand-arm system in Figure 1. The hand-arm system consists of a 7 degree-of-freedom robot arm with an attached 7 degree-of-freedom three-fingered hand. The fingers are equipped with 6 tactile arrays (2 per finger) containing 486 texels in total.

The robot's task was to determine the viscosity $\boldsymbol{\theta}$ of various liquids in bottles by shaking the containers using different actions $\mathbf{x}$. The action parameters include the shaking frequency, as well as the rotation angle of shaking (that is, the angle of the last arm joint at one of the end-points of the shaking trajectory). At a rotation angle of 0 degrees, the bottle is held horizontally while being shaken. For positive or negative rotation angles, the bottle is tilted upwards or downwards, respectively. Sensor data $\mathbf{y}$ is measured from tactile arrays mounted on the robot's fingers and is preprocessed as described in the next paragraphs.

To gather training data, we took bottles containing three different liquids and recorded the tactile responses while shaking the bottles at a range of frequencies (from 0.3 to 1.1 Hz) and rotation angles (from -25 to 25 degrees) for 5 seconds each. Training data came both from a fixed grid over action space and randomly chosen actions. The three liquids had viscosities of 1 cst (water), 120 cst (motor oil), and 1200 cst (glycerine). These values were transformed to $\log_{10}$ space, yielding values of 0, 2.07 and 3.07, respectively. The bottles used for the three liquids had identical shape, and the content was matched for weight (160g). Bottles were gripped using a force-controlled strategy, after which finger joint position were held constant throughout the shaking motions. A typical response profile during shaking movements consisted of responses from 20 texels.



(a) Mean of Gaussian process for observed power at frequency 1.5 Hz



(b) Mean of Gaussian process for observed power at frequency 1.8 Hz.

Fig. 2. Examples of learned GP mean functions over power spectrum amplitudes at different observed frequencies. Left: Water (low viscosity). Right: Glycerine (high viscosity).

We preprocessed the tactile data in the following way. First, the time series recorded during shaking movements were projected onto their principal component, in order to achieve spatial invariance with respect to the most responsive texels. Then, we calculated the Fourier transform of the resulting time series and normalized the power spectrum between the frequencies of 1.35 Hz and 2.9 Hz in order to remove any effects due to variations in individual grip strengths and locations. The time series data were re-normalized at each time step in order to combat drifting of the tactile responses over time.

We then fitted individual sparse GPs to each Fourier component (11 in total). The resulting model maps the joint space of viscosity and action parameters $[\boldsymbol{\theta}\ \mathbf{x}]^T$ to the preprocessed tactile space $\mathbf{y}$. We collected 5070 training points in total and used 297 pseudo-inputs. The pseudo-input were initialized over a grid over action parameter space (instead of random initialization) before being optimized.

Figure 2 shows slices of the learned GP's mean functions for two different output frequencies for both water (left figures) and glycerine (right figures). It can be seen that most of the power is concentrated at a frequency that is double the shaking frequency (vertical red bands). However, other frequency bands also contribute power (yellow bumps throughout plot). The specific relationship between action parameters and observed power spectrum most likely depends on the exact shape of the bottle as well as other parameters.

As can be seen from the plots, at certain shaking angles, the amplitudes between water and glycerine filled bottles differ significantly—at these angles, the two liquids could be distinguished more easily. It should be noted though, that the plots only show the mean function and not the corresponding variance, which also affects discriminability.

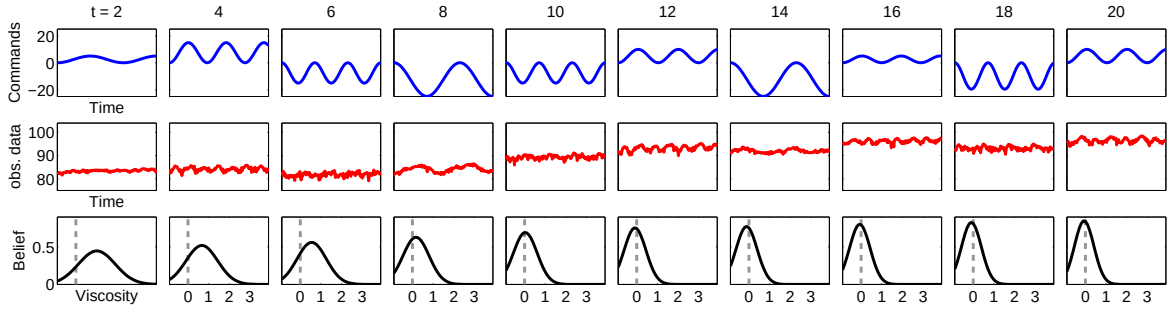
B. Performance comparison

We compared four different strategies for determining optimal actions at each time step:

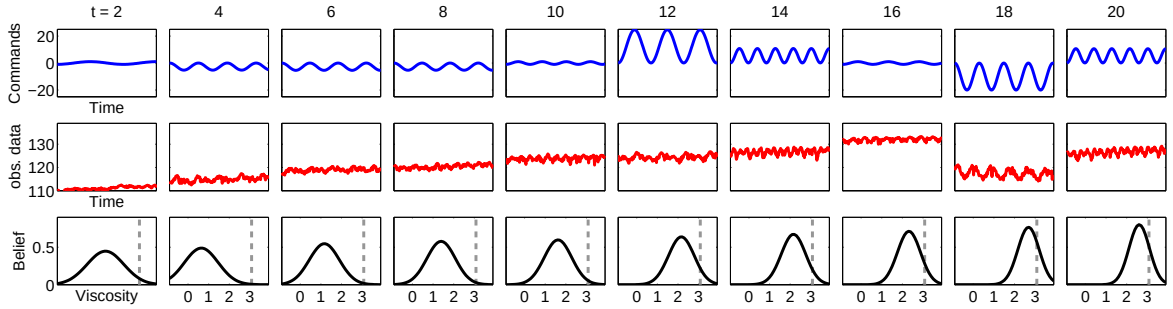
1) *Random strategy*: Actions are selected randomly from a uniform distribution over the action space that was explored in the initial training phase.
2) *Grid strategy:* A grid is placed to cover the action space uniformly. This ensures high and low frequencies, as well as a range of rotation angles, are used.
3) *Frequency strategy:* The shaking frequency is increased over time while the bottle is held constant at zero rotation angle.
4) *Active learning strategy:* The informativeness of each subsequent action is maximized by gradient ascent on the current information landscape.

At the start of run/test time, we placed an initial broad Gaussian prior over the viscosity space and ran the experiment for 20 time steps. At the very first time step, a shaking frequency of 0.5 Hz was always chosen since the sensor values had to be normalized to account for slight variations in grip force and grip location. In subsequent steps, shaking frequencies were selected according to one of the four strategies described above. As in the initial training part, we recorded tactile responses for 5 sec. When switching action parameters, we allowed the liquid to settle into the new shaking pattern for 3 sec, before starting to record new tactile observations.

For the active learning strategy, the execution time of optimization was capped to a maximum value of 500 msec at each time step in order to ensure that the optimization procedure did not affect overall run-time. Each shaking behavior/trajectory (corresponding to a set of action parameter values) needed to be completed before a new set of actions could be performed. On average, it took more than 500 msec for a specified shaking trajectory to complete.



(a) Estimation run with a bottle containing water.



(b) Estimation run with a bottle containing glycerine.

Fig. 3. Sample runs with two different liquids. Each individual column corresponds to a single time step (lasting 5 sec). Shown are the commanded joint angle of the last arm joint (top plot—where the abscissa indicates time and shows commands taken over 5 secs), average tactile observations (middle—where the abscissa again indicates time and shows sensor data observed over a period of 5 sec) and belief over log viscosity space (bottom—where the x axis represents log viscosity values ranging from -1 to 4) at each time step. Note that only every other time step t is shown in the figure.

Figure 3 shows two step-by-step sample runs for water and glycerine-filled bottles using the active learning strategy. Further examples are shown in the accompanying video.

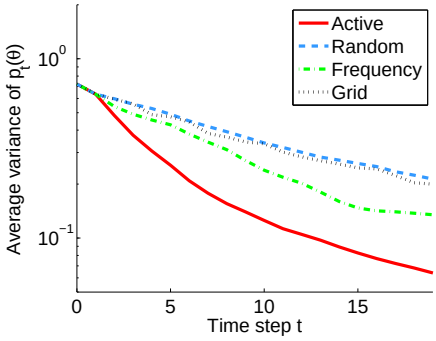


Fig. 4. Posterior variance of viscosity (on log scale) over time for active learning (red), random (blue), frequency (green) and grid (black) strategies.

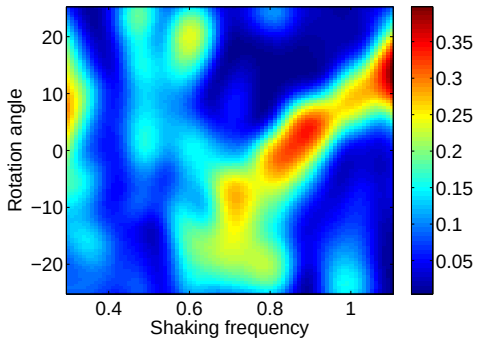
Instead of doing classification over a fixed number of liquids, our model is continuous over the viscosity domain and, therefore, is capable of exhibiting generalization properties. To evaluate model generalization, we introduced a fourth new liquid—a water-glycerine mix with a viscosity of 30 cst (1.47 in log space). We performed six trials for each liquid, using both the random and active learning strategies, and three trials for both the frequency and grid strategies. Estimation worked equally well for the liquids used in the training phase as for the newly introduced liquid.

The mean squared error—averaged over all four liquids and trials—for the active learning strategy after 20 time steps was 0.48. The frequency strategy was close with 0.52, while the other strategies had a mean squared error of 0.72 and greater. Additionally, the posterior variance after 20 steps was considerably lower for the active learning strategy than for any of the other strategies. Figure 4 shows the average posterior variance of $\boldsymbol{\theta}$ over time, indicating that the active learning strategy leads to the fastest convergence. The figure shows that the active strategy is able to reduce the viscosity's uncertainty in just 7-10 steps to the same level that the other strategies take 20 steps to achieve.

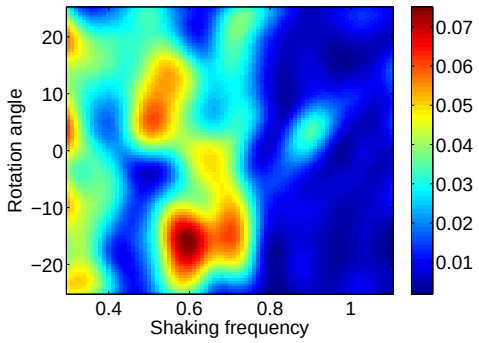
C. Information landscape

It is generally very hard to come up with information maximizing strategies by hand, since small changes in the sensor model can result in large changes in informativeness of different action parameters. Moreover, in problems where there is no analytical model of all involved effects, intuitions about what might be effective strategies can be misleading. Another challenge is that in non-linear problems, the informativeness of different actions also depends on the current belief state. Depending on what is already known about the problem, different actions may be desirable.

Figure 5 shows information landscapes for two different prior distributions over θ. As Figure 5(a) illustrates, when uncertainty about the true viscosity is high, relatively high



(a) Information landscape for a broad distribution over viscosity space ($\boldsymbol{\mu} = 1.5$ and $\boldsymbol{\Sigma} = 0.7$).



(b) Information landscape for narrow distribution in a low viscosity region ($\boldsymbol{\mu} = 0.3$ and $\boldsymbol{\Sigma} = 0.2$).

Fig. 5. Information landscapes for two different probability distributions of θ (a broad one centered over a high viscosity value of 1.5 and a narrow one centered over a low viscosity value of 0.3). Increasing shaking frequencies are shown on the abscissa, different rotation angles on the ordinate. Blue regions have low information, while yellow and red regions are medium and highly informative, respectively.

shaking frequencies are preferred, along with a positive rotating angle (resulting in an upwards tilted bottle) that increases with shaking frequency. On the other hand, when the distribution over θ is restricted to a narrow low viscosity region, low shaking frequencies become much more informative, as Figure 5(b) shows. This trend can also be seen in Figure 3, where towards the end of each run, low shaking frequencies are selected for the water-filled bottle (see Figure 3(a)), while high shaking frequencies are chosen for the glycerine-filled bottle (see Figure 3(b)).

Since informative regions of the action space can vary drastically with the current belief and highly informative regions tend to be sparse, any strategy relying on chance to encounter such informative regions could be expected to fail. Moreover, regions of high information may be located very close to parameter regions containing very little information. As a result, small differences in action values can have a huge effect on how informative resulting observations will be.

V. DISCUSSION

We presented an active framework that exploits a learned sensor model in order to determine dynamics parameters of objects. We derive a model based on sparse Gaussian Processes and evaluate our framework on a real robotic hand-arm system that is able to determine the viscosity of different liquids by shaking bottles at different frequencies and angles. Optimal actions are performed at each time step by so that most informative observations (i.e., informative with respect to the liquid's viscosity) are gathered. We demonstrated that the active learning strategy performs better than other simple strategies and is not slowed down by protracted calculations.

An interesting extension of this work would be to incorporate active learning concepts to the collection of the training data (in addition to run-time data). As the dimensionality of the action space increases, the amount of training data samples grows very quickly, and collecting training data can be very time consuming and involved. One can imagine gathering data only from regions that would contribute to a sufficiently rich training set so that the sensor model can be properly learned in a reasonable amount of time.

ACKNOWLEDGMENTS

This work was supported by the EU project SENSOPAC (IST-028056) and an EPSRC DTC scholarship to H. P. S.

REFERENCES

[1] E. Snelson and Z. Ghahramani, "Sparse Gaussian processes using pseudo-inputs," in *NIPS*. MIT Press, 2006.

[2] H. P. Saal, J. Ting, and S. Vijayakumar, "Active sequential learning with tactile feedback," in *Proc. 13th Int. Conf. on Artificial Intelligence and Statistics (AISTATS), JMLR: W&CP*, vol. 9, 2010.

[3] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Rigid body load identification for manipulators," in *Proc. 24th Conf. on Decision and Control*, vol. 24, 1985, pp. 996–1002.

[4] S. Takamuku, G. Gomez, K. Hosoda, and R. Pfeifer, "Haptic discrimination of material properties by a robotic hand," in *Proc IEEE Int. Conf. on Development and Learning (ICDL)*, 2007.

[5] M. Suzuki, K. Noda, Y. Suga, T. Ogata, and S. Sugano, "Dynamic perception after visually guided grasping by a human-like autonomous robot," *Adv Robotics*, vol. 20, no. 2, pp. 233–254, 2006.

[6] S. Takamuku, K. Hosoda, and M. Asada, "Object category acquisition by dynamic touch," *Adv Robotics*, vol. 22, no. 10, pp. 1143–1154, 2008.

[7] J. Denzler and C. M. Brown, "Information theoretic sensor data selection for active object recognition and state estimation," *IEEE T Pattern Anal*, vol. 24, no. 2, pp. 145–157, 2002.

[8] J. Bongard, V. Zykov, and H. Lipson, "Resilient machines through continuous self-modeling," *Science*, vol. 314, no. 5802, pp. 1118–1121, 2006.

[9] L. Paletta and A. Pinz, "Active object recognition by view integration and reinforcement learning," *Robot Auton Syst*, vol. 31, no. 1–2, pp. 71–86, 2000.

[10] E. Tuci, G. Massera, and S. Nolfi, "Active categorical perception in an evolved anthropomorphic robotic arm," in *Proc. IEEE Congress on Evolutionary Computation*, 2009, pp. 31–38.

[11] C. K. I. Williams and C. E. Rasmussen, "Gaussian processes for regression," in *NIPS*. MIT Press, 1995.

[12] W. Liu, I. Park, and J. C. Principe, "An information theoretic approach of designing sparse kernel adaptive filters," *IEEE Trans Neural Netw*, vol. 20, no. 12, pp. 1950–1961, 2009.

[13] R. Martinez-Cantin, N. de Freitas, E. Brochu, J. Castellanos, and A. Doucet, "A bayesian exploration-exploitation approach for optimal online sensing and planning with a visually guided mobile robot," *Auton Robot*, vol. 27, pp. 93–103, 2009.

[14] M. P. Deisenroth, M. F. Huber, and U. D. Hanebeck, "Analytic moment-based gaussian process filtering," in *ICML*, 2009.

[15] S. Julier and J. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[16] A. Girard, C. E. Rasmussen, J. Q. Candela, and R. Murray-Smith, "Gaussian process priors with uncertain inputs – application to multiple-step ahead time series forecasting," in *NIPS*. MIT Press, 2003, pp. 545–552.