INTUITIVE

INnovative Network for **T**raining in To**U**ch **I**nterac**TIVE** Interfaces

Grant agreement: #861166 H2020-MSCA-ITN-2019

Start date: 2021-05-01 End date: 2024-04-30

Delivarable reporting document

Deliverable no: 5.3		WP: 5
Deliverable name: Haptic integration of proprioception and tactile information for robotics	Type: Report	Dissemination level: Public
Due delivery date: 30 Nov 2023		Date delivered: 5 Dec 2023

Description: The deliverable presents details, formulation and implementation of a novel methodology based on "predictive coding" to integrate visual, haptic and proprioception for robotic manipulators.

Contents

1 Introduction

Robotic manipulators are increasingly used in novel and unstructured environments for strong physical interaction with the environment. For such contact-rich manipulation tasks, it becomes critical to know the object's geometry, mass, surface friction coefficient, stiffness, etc. Acquiring knowledge about these properties of previously unperceived objects could help to achieve a stable and accurate manipulation [1–3] and also help to predict the effects of various manipulation actions in advance.

The combination of tactile or 'touch' sensing with spatial kinesthetic information forms the terminology of 'haptics' [4]. For robotic manipulators, kinesthetic information (proprioception) provides accurate information about the manipulator's state, i.e., how it is moving in space. This has been extensively researched and well defined under forward kinematics and dynamics of serial robotic manipulators. On the contrary, tactile perception embodies the outcome of actions taken and depends on the properties of objects engaged by the robotic manipulator. This fundamental principle has been considered as the basis for this research work. Furthermore, obtaining an informative tactile perception requires meaningful interaction or actions, which transforms it into a challenge for interactive perception [5]. This interactive perception problem is presented in Figure 1, where two key research questions are explored to address the problem of the combination of proprioception and tactile information in Deliverable 5.3: i) How to efficiently represent the object and ii) How to take action that provides meaningful sensory information.

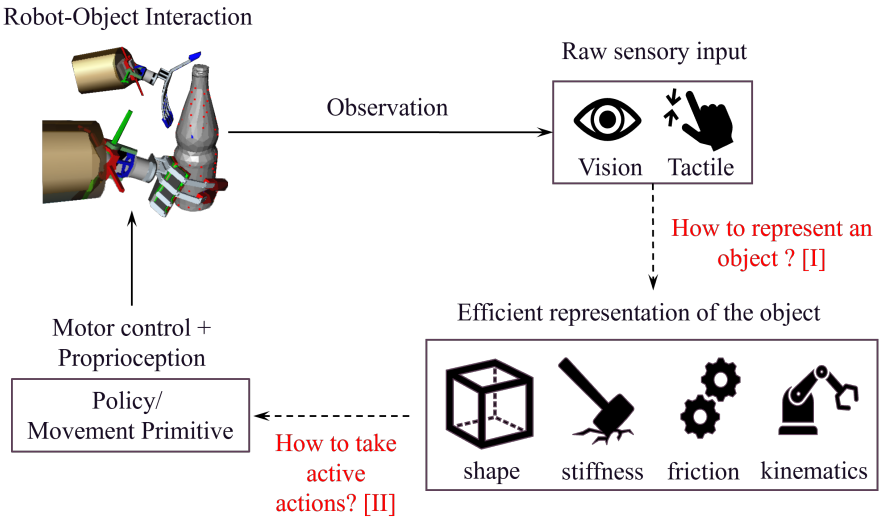

Figure 1: Object property inference through interaction

In addition to combining proprioception with tactile sensing, we sought to combine vision and haptic sensing to develop a generalizable robotic exploratory approach. The importance of vision-based perception and haptics is well established in robotics. Vision is crucial to obtain a global overview and often guides haptic exploration, making it necessary to combine complementary modalities [6, 7]. There has been overwhelming work done in the perception of objects and environments based only on vision and cameras. However, the possible range of properties that can be observed using vision is quite limited. In this regard, touch or tactile sensing has been considered a crucial sensing modality that can provide a rich and diverse set of information about the object

with which the robotic system interacts [8, 9]. However, most of the methods developed for haptic perception are limited to passive classification or recognition problems [9]. In addition, they employ bottom-up processing, limiting the application to known scenarios and short-term utilization.

To approach such a challenging problem, it might be interesting to take inspiration from research in neuroscience, where such behavior is baked into the human cognitive system. To manipulate novel objects, humans often perceive object properties through actions such as holding, grasping, or pushing to gain better control [10, 11]. In such *interactive visuo-tactile perception*, active physical interactions or explorations are made to improve object perception [5, 12–14]. A key working principle of such human perception that has been flourishing for many decades is that humans actively predict perception via learning an internal model of the environment. This has recently been formalized under 'Predictive Coding' and 'Active Inference [15, 16].

In this research project, such a 'predictive' framework is developed for robotic manipulation systems to infer object properties through contact-rich interaction. The key aspect of the framework will be encoding contact-rich interactions as Probabilistic Markov Models and learning the interaction model. The learned interaction model will be utilized to infer the physical parameters like mass, the center of mass, relative friction, etc. using the Bayesian inference scheme. The developed framework will be a step towards developing generalizable robotic manipulation exploratory skills and will be crucial to performing downstream robotic manipulation tasks efficiently.

2 Related Works

In this section, a brief overview of previous works is presented which addressed the problem of exploring the properties of unknown objects. Estimating the physical properties of novel objects is a challenging problem in robotics, using either vision or tactile sensing. Physical object properties are not salient in static or quasistatic interactions, and often each parameter is only revealed in specific interactions, making it an interesting research problem [17].

One of the earliest works of Atkeson et al. [18] estimated the mass and moment of inertia of an object rigidly attached to a manipulator, using joint torques and a wrist-mounted force-torque sensor. Similar results have also been presented in [19]. These approaches required the object to be manually attached to the end-effector. Few works elevated this constraint as in [20], where the authors used a custom 2-finger mechanism to measure contact forces during planar push and in [21], the authors applied a tilt approach to measure wrenches to estimate inertial parameters. Zhao et al. [22] incorporated friction estimation, by grasping the object and measuring the contact forces during the sliding regime. Most of these prior estimation techniques relied on precise force or tactile sensing, assumptions about the object geometry, or the interaction between the object and environment, and employed specialized mechanisms, thus making it difficult for generalization and autonomous exploration of the object.

Some researchers attempted to overcome the limitations mentioned above by introducing interactive manipulation techniques such as grasping or pushing. In [23], the authors

estimated only the mass of an object by controlled push, which required prior knowledge of the friction coefficient of the surface. Similarly, to determine the center of mass of the object, Yao et al. [24] used tactile forces during a 3-fingered robotic grasp. To estimate a wide range of physical properties of the object, Sundaralingam et al. [25] used a factor graph approach using in-hand manipulation with precise tactile and force-torque sensing. The approach relied on the approximation of in-hand object dynamics, known object shape, and a marker-less tracking system. More recently, Uttayas et al. [**?**] estimated viscoelastic properties using a filtering approach, based on an approximate spring mass damper model. The works mentioned above employing interactive manipulation often used an analytical formulation to model the object-robot interaction, which is often approximate and has significant assumptions about the interactions.

Recently, data-driven and physics engine approaches are being taken to overcome such problems. Wu et al. [26] used deep learning to learn interactions between objects colliding in a physics engine and used the learned model to estimate the mass and friction parameters for real object motion. Song et al. [27], [28] relied on a physics engine to predict expected object motions during pushing and employed Bayesian optimization on a real object motion to predict distributed mass and friction on objects. These works often relied on the accuracy of the physics engine and are generally computationally complex. Xu et al. [17] used only vision and deep learning to learn a representation of the mass and friction coefficient by randomly pushing and poking objects. Mavrakis et al. [29] collected large pushing trajectories (40k) in the simulation environment and learned a regression model for estimating an object's inertial parameters during non-prehensile pushing. However, these approaches require intensive training and do not involve strategic interaction. In this work, we propose an active formulation for efficient training data-driven object-robot interaction model.

Until now, either vision or tactile were used to estimate the physical object properties. On one hand, tactile information is crucial to infer multiple object properties like in [24, 25, 30, 31], however, it requires precise position information and prior knowledge. On the other hand, vision-based approaches such as [17, 32] could only estimate fewer object properties with higher error rates, but required no prior knowledge about the object. To exploit the complementing vision and tactile sensing modalities, we propose to utilise both. Recently, Murali et al. [33] and Lee et al. [34] have shown visuo-tactile based approach significantly improves the performance of the robotics systems problems like pose estimation and contact-rich manipulation.

To tackle the above-aforementioned problems and constraints for estimating or inferring the physical object properties, we present the proposed approach in the following section. We present extensive experiments to validate our approach and compare with a *non-predictive* state-of-the-art method to demonstrate the advantage of using predictive formalism.

3 Research Approach

In this section, the problem of active visuo-haptic object inference under the 'predictive' framework is formally introduced and an overview of the proposed research approach is presented. Humans learn an intuitive understanding of contact-rich interactions through

playful interaction [35, 36], and then use the learned interaction dynamics to explore and infer the inherent properties of novel objects. In addition, they utilize both vision and haptics in a complementary manner. Motivated by such exploration, in this work, we focus on learning a key contact-rich interaction model for robotic manipulation, non-prehensile manipulation. Non-prehensile manipulation with objects is difficult to formulate analytically due to the contact dynamics, non-linearity and discontinuous behavior. As presented in Section 2, inspired by the recent progress in data-driven model learning, learning such intricate models could lead to better generalization and robustness. After learning the interaction model, we actively infer the essential properties of the previously unseen objects.

3.1 Problem Formulation

We consider the problem of estimating the state s of an unknown rigid object from vision o^V and tactile observation o^T using non-prehensile pushing actions a. At any given time t, the state $s_t = \{\psi_t, \phi\}$ comprises of time-varying factors: **pose**, $\psi_t = \{x_t, y_t, \theta_t\}$ i.e. how the object is moving in the table, as well as time-invariant factors: **parameters**, $\phi = \{m, \mu, CoM_x, CoM_y, I_z\}$ as *mass, relative friction coefficient between object and surface, center of mass, inertia*. The center of mass is measured w.r.t. frame attached to the geometric center of an object and only the rotational inertia in 2D is considered, as the interaction is restricted to motion in 2D. Observation o_t^V consists of RGB-D images of the pushing area and tactile observation o_t^T consisting of *2D contact forces, contact indicator*. The contact indicator $\in 0, 1$, depending on whether the robot and the object are in contact. The pushing action a_t is parameterized by the *contact point* (cp), *push direction* (pd) and *velocity* (v) of the push. cp consists of the 2D world coordinate of the contact point, pd, the rotational angle of the z-axis of the robotic system aligned along a pushing direction & v is the magnitude of push velocity by the robotic system.

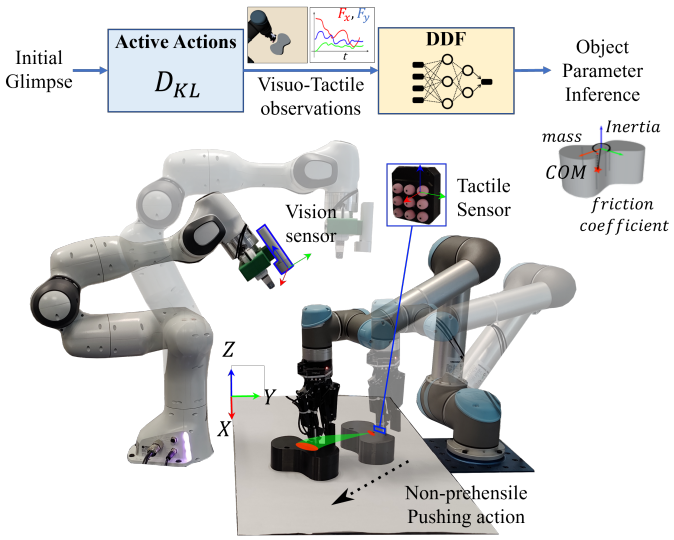

Figure 2: Problem setup for visuo-tactile based active object parameter inference

We perform quasi-static pushing [37] to infer the object parameters ϕ which are not directly observable either through vision or tactile sensing. The change in **pose** of the object over time depends not only on how the interaction action was taken but also on the

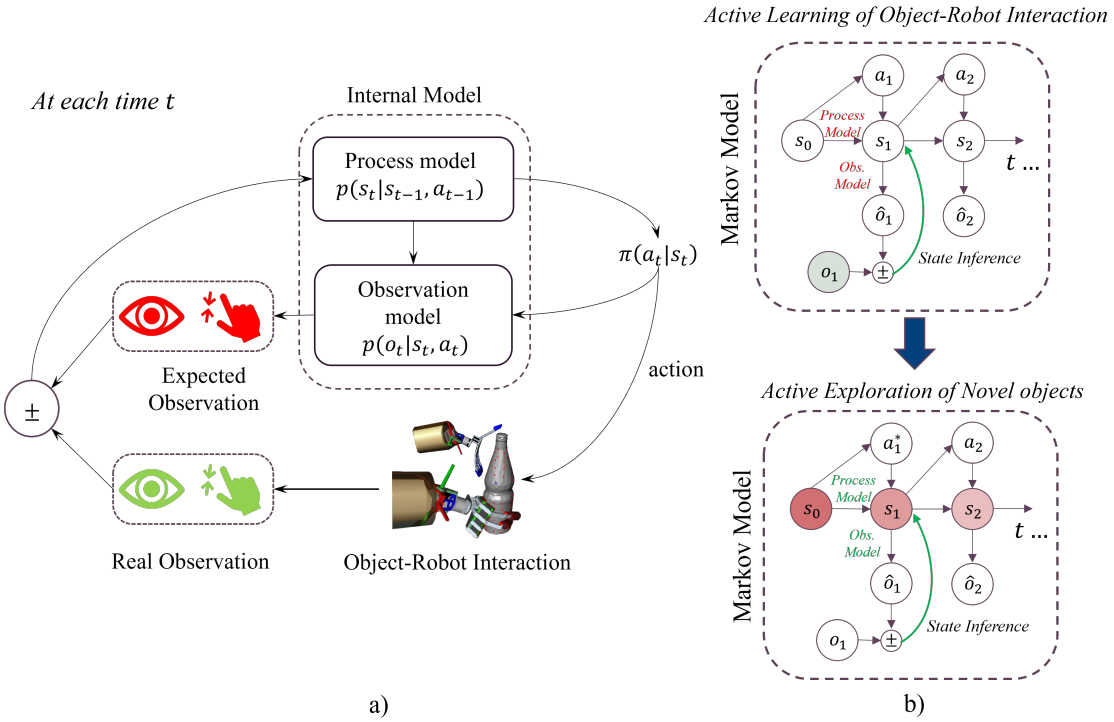

Figure 3: Proposed 'predictive processing' framework. Part (a) presents how the internal model is utilized to generate an expected observation which is then compared with actual observation to update the internal model. Part (b) presents the Markov Chain Model of both learning and inference for multiple time steps.

properties. In this setting, the object state, especially the parameters of the objects, is not directly observable. Thus, the problem falls under a partially observable Markov Decision Process and can be encoded as a Markov chain. The key aspect of the research approach is to address the problem via a human-inspired 'predictive processing' framework. After an extensive review of the literature, the essential takeaway of such a framework which applies to the problem at hand is learning and inferring using an internal model of the environment in the Bayesian Inference setting, which is depicted in Figure 3. We present a complete formulation of the proposed 'predictive' processing framework.

3.2 Bayesian Inference

In Bayesian inference settings, the knowledge about the current state of the object s_t is represented with a distribution conditioned on previous actions $a_{1:t}$ and observations $o_{1:t}$. This distribution is denoted as $bel(s_t)$

$$bel(s_t) = p(s_t|o_{1:t}, a_{1:t}) \tag{1}$$

To compute the belief, the Bayes rule has to be employed for inferring the state

$$p(s_t|o_{1:t}, a_{1:t}) = \frac{p(o_t|s_t, o_{1:t-1}, a_{1:t})p(s_t|o_{1:t-1}, a_{1:t})}{p(o_t|o_{1:t-1}, a_{1:t})} \tag{2}$$

However, the denominator in Eq.2 is intractable to compute, as it requires a distribution over all possible observations which is not possible to know beforehand.

Various approaches exist to perform such Bayesian inference tractably, such as Variational Inference, Recursive Bayesian Filters, and Expectation-Maximization [38]. In addition, learning the internal model (process and observation) is often not part of the inference formulation. Learning and possessing an internal generative model of the complete world like the human brain is too complex, and on top of it performing inference on it. The active inference or variational autoencoder-based approach presented by [16], overcomes such a challenge. However, as presented in Section 2, the application of the high-dimensional and continuous domain of object property inference is quite limited.

One potential approach to such a scenario is to employ Recursive Bayesian Filters. Recursive Bayesian filters are a popular class of algorithms that enables the computation of this distribution in cases where the observations follow Markov assumptions and are conditionally independent. Kalman Filters are a popular choice and are optimal for linear systems. Additionally, recent work [39] demonstrated that under the Gaussian assumption, the Active Inference optimization of the Free Energy principle is the same as that of Kalman-optimal filtering.

$$bel(s_t) = p(s_t|o_{1:t}, a_{1:t}) = \eta\, p(o_t|s_t, a_t)\overline{bel}(s_t) \tag{3}$$

$$\overline{bel}(s_t) = \int p(s_t|s_{t-1}, a_{t-1})bel(s_{t-1})ds_{t-1} \tag{4}$$

Two key aspects of such a Bayesian Filter is the representation of the internal model as separated into two components - the forward model or process model of the state in form of $p(s_t|s_{t-1}, a_{t-1})$ which captures the evolution of the internal states and an observation model relating the states to the observations $p(o_t|s_t)$ which relates how the observation relates to the internal states. In addition, both have associated noise models that reflect the stochasticity of the underlying system and determine how much trust the filter places in the process and observation models. To have a seamless connection of Recursive Bayesian Filtering to the Active Inference formulation a variational outlook is presented.

From Eq.2.

$$p(s_t|o_{1:t}, a_{1:t}) = \frac{p(o_t, s_t|o_{1:t-1}, a_{1:t})}{p(o_t|o_{1:t-1}, a_{1:t})} \tag{5}$$

Taking log on both sides and re-arranging-

$$\begin{aligned} -ln(p(o_t|o_{1:t-1}, a_{1:t}) &= ln(p(s_t|o_{1:t}, a_{1:t})) \\ &\quad -ln(p(o_t, s_t|o_{1:t-1}, a_{1:t}) \end{aligned} \tag{6}$$

Adding approximate variational distribution $q(s_t) \sim p(s_t|o_{1:t}, a_{1:t})$

$$\begin{aligned} -ln(p(o_t|o_{1:t-1}, a_{1:t}) &= ln(p(s_t|o_{1:t}, a_{1:t})) \\ -ln(p(o_t, s_t|o_{1:t-1}, a_{1:t}) + ln(q(s_t)) &- ln(q(s_t)) \end{aligned} \tag{7}$$

Integrating and simplifying results in

$$-ln(p(o_t|o_{1:t-1}, a_{1:t}) = F - \mathbf{D}_{KL}(q(s_t)||p(s_t|o_{1:t}, a_{1:t})) \tag{8}$$

where F is the free energy term or evidence lower bound.

$$F = \mathbb{E}_{q(s_t)}[ln(q(s_t)) - ln(p(o_t, s_t|o_{1:t-1}, a_{1:t}))] \tag{9}$$

In recent works [40], researchers have attempted to perform state estimation via Stochastic Gradient descent on the Free Energy Term F, w.r.t sufficient statistics of the state distribution. However, under the Gaussian setting as ours $q(s_t) \sim \mathcal{N}(s_t; \mu_{s_t}, \Sigma_{s_t})$ and mean-field approximation, the gradient descent objective is results are same optimization function as that of Kalman Filter as stated above.

The Free Energy formulation is extended to infer actions under the active inference scheme. Consider future time steps $t = \tau, ..\tau + T$. An expected free energy term is introduced wherein the variational distribution is conditioned on future actions sequences $\pi = a_{\tau:\tau+T}$ and a biased distribution $\tilde{p}(.)$ is introduced to encode preference over future goal states or observations. The Expected Free Energy (EFE) G for a sequence of action π at each time step is given by-

$$G_\tau(\pi) = \mathbb{E}_{q(s_\tau|\pi)}[ln(q(s_\tau|\pi) - ln(\tilde{p}(o_\tau, s_\tau|\pi)] \tag{10}$$

This can be further approximated and decomposed as-

$$\begin{aligned} G_\tau(\pi) \approx -\mathbb{E}_{q(s_\tau|\pi)}[ln(\tilde{p}(o_\tau)] \\ -\mathbb{E}_{q(s_\tau|\pi)}[ln(q(s_\tau, o_\tau|\pi) - ln(q(s_\tau|\pi)] \end{aligned} \tag{11}$$

The Expected Free Energy is a central quantity in the theory of active inference. It is the quantity that all active inference agents are mandated to minimize through action, and its decomposition into extrinsic and epistemic value terms provides a way to exploit and explore effectively. In the problem of the project, the epistemic value can be utilized for driving active learning and active exploration to infer the unknown parameters which is presented in the following section.

3.3 Active Differentiable Filters

We employ the Kalman filtering approach for recursive Bayesian filtering. For our problem, we employ a data-driven approach to learn the process and observation model along with the respective noise models, end-to-end using a differentiable filter. Recently, differentiable filters that integrate Bayesian Filtering with deep learning [41–44] were proposed. The authors have also shown that this approach performs better compared to the standard deep-learning approach in handling real world noise and in [45] showed the strength of such an approach for a variety of tasks like visual optometry, visual object tracking, etc.

In our problem, the data-driven models within a differentiable filter capture the complex and stochastic object-robot internal interaction model during non-prehensile pushing. In addition, based on the current action (proprioception) and the previous state, it can predict tactile information, thus fulfilling the primary objective of Deliverable 5.3. Further, as the pose of the object is intricately dependent on the parameters, a straightforward combined (joint) filtering for pose and parameter does not perform well. Therefore, we utilize a dual filter design, exploiting the dependency among the states for consistent filtering and inferring the parameters of the object. In addition, learning such models and also inference is often data-intensive. To make the framework more viable for real-life scenarios, it is crucial to incorporate active learning and inference schemes, therefore motivated by

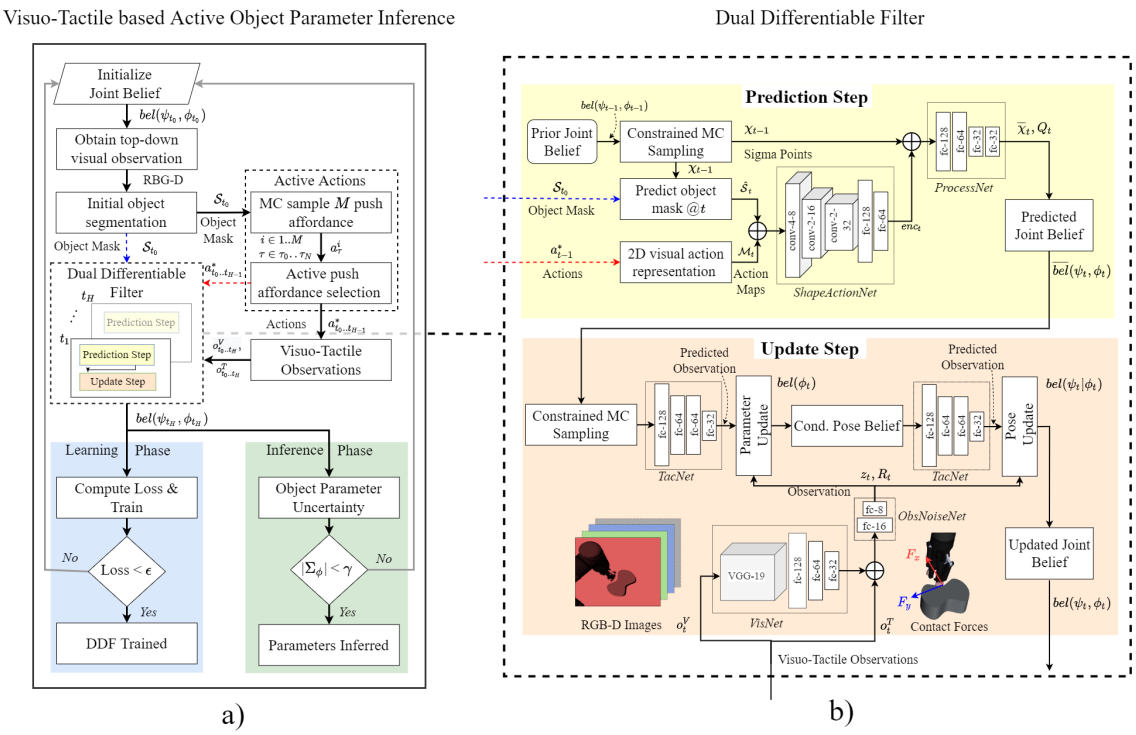

Figure 4: Our proposed framework (ADDF) for visuo-tactile based active object exploration using non-prehensile manipulation. Part a) presents the overall framework and part b) presents an expanded view of the dual differentiable filter block.

Eq.11, active action selection is formulated under such a differentiable filter setting. The proposed Active Differentiable Filter-based approach was recently implemented and was accepted for publication at the **IEEE International Conference on Intelligent Robots & Systems 2023** [46]. Our proposed approach is presented in the Figure 4. It comprises a novel dual differentiable filter for parameter and pose estimation along with data-driven models. The action selection for the push affordance is performed by computing the N step information gain term, making it an active dual differentiable filter (ADDF). Firstly, the robotic system learns the data-driven models used within the differentiable filter. After learning, we perform inference on novel and unknown objects to estimate their parameters, without prior information about the novel object. In the following sections, we explain the various components of the framework.

3.3.1 Dual Differentiable Filter

We derive our dual filter based on differentiable UKF [45, 47]. For the dual filter formulation, we explicitly represent the state s_t by the joint distribution of ψ_t and ϕ_t, via Multivariate Gaussian distribution:

$$bel(\psi_t, \phi_t) \sim \mathcal{N}(\psi_t, \phi_t | \mu_t, \Sigma_t) \tag{12}$$

with statistics $\mu_t \in \mathbb{R}^8$ and $\Sigma_t \in \mathbb{R}^{8x8}$ as

$$\mu_t = \begin{pmatrix} \mu_{\psi_t} \\ \mu_{\phi_t} \end{pmatrix}, \quad \Sigma_t = \begin{pmatrix} \Sigma_{\psi_t} & \Sigma_{\psi_t \phi_t} \\ \Sigma_{\phi_t \psi_t} & \Sigma_{\phi_t} \end{pmatrix} \quad . \tag{13}$$

The dual filter as shown in Figure 4(b) follows, the structure of a Kalman Filter with a *prediction step* and an *update step*, with key novelty explained in this section.

Prediction Step

In prediction step, the next step is the joint belief given the previous belief and the actions. The object parameters are real physical quantities with some physical constraints (for, e.g. $m, \mu > 0$). However, simply constraining the sigma points χ^{UT} in the UKF approach does not preserve the true variance of the Gaussian distribution [48]. Therefore, we perform constrained Monte Carlo sigma point sampling to preserve the physical constraints and the Gaussian variance. We employ a differentiable sampling method [49] to sample C sigma points in the joint distribution $bel(\psi_{t-1}, \phi_{t-1})$ instead of using standard Unscented Transform points:

$$\chi_{t-1}^{[i]} = \mu_{t-1} + \varepsilon^{[i]} \sqrt{\Sigma_{t-1}} \tag{14}$$

where, $i \in 1..C$ and $\chi_{t-1} = [\chi_{\psi_{t-1}}, \chi_{\phi_{t-1}}] \in \mathbb{R}^{Cx8}$ with an associated weight $w_t^{[i]} = 1/C$ and $\varepsilon^{[i]} \sim \mathcal{N}(0,1)$. We set $C = 100$ for all our experiments. The sigma points are filtered on the basis of whether they satisfy the physical constraints and passed through the data-driven models. However, the invalid sigma points are also retained and reintroduced to preserve the uncertainty of the distribution. This is visually illustrated and explained in Figure 5.

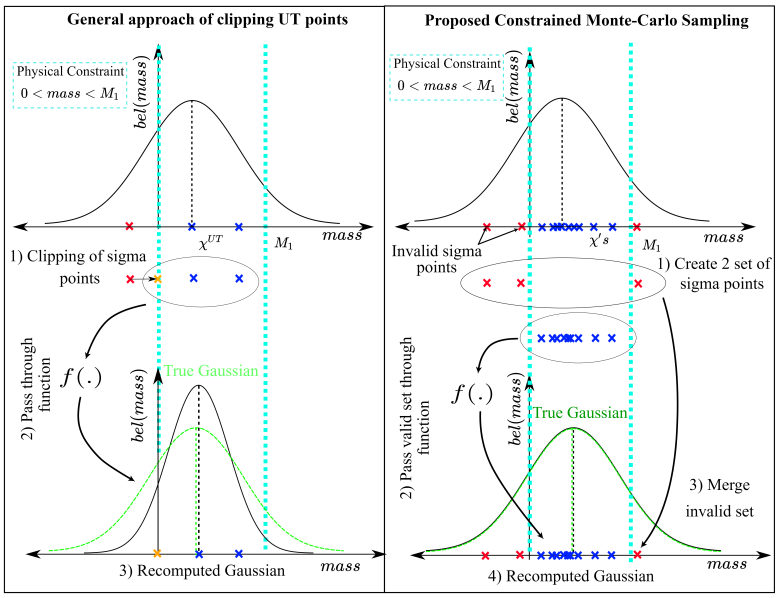

Figure 5: Constrained Monte Carlo sampling

Shape-Action Encoder: During pushing, it is important to take into account the local geometry of the object and the action. Few previous works [32, 50] have shown that such an approach improves the prediction of the action effect in form of tactile prediction. We encode the action along with the local geometry of the object at the point of contact to improve the predictions of the action effect via *ShapeActionEncoderNet*.

ShapeActionEncoderNet: This comprises 3-layer CNN layers followed by 2 layers of feed-forward neural network. For each sigma point sampled from the current belief, an expected *segmentation mask* $\hat{\mathcal{S}}_t$ is generated by transforming the initial segmentation $\mathcal{S}_0$ based on the pose information $\chi_{\psi_{t-1}}$. This represents the current geometry of the object at the point of action. Next, a 2D representation of the action - *action map* $\mathcal{M}_t$ is generated. This is done by representing a 2D Gaussian distribution based on the action affordance

$a_t = (cp, pd, v)$ in the image frame. A 2D Gaussian can be generated by the following equations, along the push direction pd and centered at the contact point in the image frame (cp_{if}).

$$px = \begin{pmatrix} pixel_x \\ pixel_y \end{pmatrix}, K = \begin{pmatrix} \frac{cos^2(pd)}{2v^2} + \frac{sin^2(pd)}{2} & \frac{sin(2pd)}{4v^2} - \frac{sin(2pd)}{4} \\ \frac{sin(2pd)}{4v^2} - \frac{sin(2pd)}{4} & \frac{sin^2(pd)}{2v^2} + \frac{cos^2(pd)}{2} \end{pmatrix}$$

$$\mathcal{M}_t = e^{(-\frac{1}{2}(px-cp_{if})K(px-cp_{if})^T)} \tag{15}$$

By this approach, we avoid generating complex object shape predictions for an intricate visual perspective, and the action maps serve as an attention mechanism to aid in learning interaction of the shape with action. A visualization of the action maps is presented in Figure 6.

ProcessNet: The data-driven process model for predicting the change in *pose* of the object is approximated via 3 layer feed-forward neural network given prior joint sigma points and the shape-action encoding enc_t. In addition, we also employ the learnt heteroscedastic process noise model.

$$enc_{a_t} \longleftarrow ShapeActionEncoderNet([\hat{\mathcal{S}}_t, \mathcal{M}_t]) \tag{16}$$

$$\overline{\chi}_{\psi_t}, Q_t \longleftarrow ProcessNet(\chi_{t-1}, enc_{a_t}) \tag{17}$$

$$\overline{\chi}_{\phi_t} = \chi_{\phi_{t-1}} \tag{18}$$

where, $Q_t \in \mathbb{R}^{Cx3}$ is the heteroscedastic diagonal covariance noise for time-varying pose. The predicted next step sigma points $\overline{\chi}_t$, along with the process noise Q_t are utilized to compute the expected Gaussian belief $\overline{bel}(\psi_t, \phi_t)$ as

$$\overline{\chi}^{[i]}_{\psi_t} = \overline{\chi}^{[i]}_{\psi_t} + \varepsilon^{[i]}\sqrt{Q^{[i]}_t} \tag{19}$$

$$\overline{\mu}_t = \sum_{i=0}^{C} w^{[i]}_t \overline{\chi}_t \tag{20}$$

$$\overline{\Sigma}_t = \sum_{i=0}^{C} w^{[i]}_t (\overline{\chi}^{[i]}_t - \overline{\mu}_t)(\overline{\chi}^{[i]}_t - \overline{\mu}_t)^T \tag{21}$$

where, $i \in 1..C$ and $\overline{\chi}_t = [\overline{\chi}_{\psi_t}, \overline{\chi}_{\phi_t}]$

Update Step

We recompute the constrained Monte-Carlo sigma point $\overline{\chi}'_{\phi_t}$ sampling on the predicted belief $\overline{bel}(\psi_t, \phi_t)$ to incorporate the noise from the process. The dual filter employs a separate update of parameter belief similar to the parameter update presented in [51] and the conditional pose belief update based on the UKF update [47]. For updating the joint belief, we require an observation model to predict the observation sigma points $\overline{\mathcal{Z}}_t$ that must account for visual and tactile observations. To reduce the complexity of predicting raw RGB-D images, we split the observation model into two components, tactile and visual models. A *VisNet* network acts as a synthetic sensor generating the current noisy 2D pose information x, y, θ from the current RGB-D images at each time. The *VisNet*

comprises of first 10 layers of VGG-19 [52] pre-trained on ImageNet followed by 3 layers of feed-forward network. For the tactile counterpart, a 4 layers of feed-forward network *TacNet* is utilized to predict the contact force information (tactile). In addition, a two-layer network *ObsNoiseNet* is also used to generate heteroscedastic and diagonal observation noise.

$$\overline{\mathcal{Z}}_t^V = \overline{\chi}'_{\psi_t} \overline{\mathcal{Z}}_t^T \longleftarrow TacNet(\overline{\chi}'_t, enc_{a_t}) \tag{22}$$

$$z_t^V \longleftarrow VisNet(o_t^V), z_t^T = o_t^T \tag{23}$$

$$R_t \longleftarrow ObsNoiseNet(z_t^V, z_t^T) \tag{24}$$

Parameter Update We update the weights based on the likelihood of the observation sigma points $\overline{\mathcal{Z}}_t = [\overline{\mathcal{Z}}_t^T, \overline{\mathcal{Z}}_t^V]$ in the observation distribution $\sim \mathcal{N}(.|z_t, Q_t)$

$$w_t^{[j]} = w_t^{[j]} e^{(-\frac{1}{2}(\overline{\mathcal{Z}}_t^{[j]} - z_t)R^{-1}(\overline{\mathcal{Z}}_t^{[j]} - z_t)^T)} \tag{25}$$

where $j \in 1..C$. The updated parameter belief $bel(\phi_t)$ is recomputed via a Gaussian Smooth Kernel [51] method after normalizing the updated weights.

$$\mu_{\phi_t} = \sum_{i=0}^{C} w_t^{[i]} \overline{\chi}'_{\phi_t}; \quad m_{\phi_t}^{[i]} = a\overline{\chi}'_{\phi_t} + (1-a)\mu_{\phi_t} \tag{26}$$

$$\Sigma_{\phi_t} = h^2 \sum_{i=0}^{C} w_t^{[i]} m_{\phi_t}^{[i]} - \mu_{\phi_t} \tag{27}$$

where a and $h = \sqrt{1-a^2}$ are shrinkage values of the kernels that are user-defined and set to 0.01, and m are the kernel locations.

Pose Update We make use of the dependence of the pose on the parameters to compute the conditional pose distribution $bel(\psi_t|\phi_t) \sim \mathcal{N}(\psi_t|\mu_{\psi_t|\phi_t}, \Sigma_{\psi_t|\phi_t})$ using Multivariate Gaussian Theorem [53].

$$\mu_{\psi_t|\phi_t} = \psi_t + \Sigma_{\psi_t\phi_t}\Sigma_{\phi}^{-1}(\phi_t - \mu_{\phi_t}) \tag{28}$$

$$\Sigma_{\psi_t|\phi_t} = \Sigma_{\psi_t} - \Sigma_{\psi_t\phi_t}\Sigma_{\phi_t}^{-1}\Sigma_{\phi_t\psi_t} \tag{29}$$

For conditional pose update, standard Unscented Kalman Filter (UKF) is employed on the predicted conditional pose distribution $\overline{bel}(\psi_t|\phi_t = \mu_{\phi_t})$ using Eq.29. The μ_{ϕ_t} of the updated parameter belief is utilised with predicted pose sigma points $\overline{\chi}_{\psi_t}^{UT}$ to obtain the predicted observation sigma points $\overline{\mathcal{Z}}'_t$. The UKF update equations are skipped for brevity. After the conditional pose update, the posterior joint is computed as:

$$bel(\psi_t, \phi_t) = bel(\psi_t|\phi_t)bel(\phi_t) \tag{30}$$

Note, the cross-covariance matrices $\Sigma_{\psi_t\phi_t}, \Sigma_{\phi_t\psi_t}$ are not updated through the dual update step and are kept constant.

3.3.2 Active Actions

The push action affordance is given by the tuple $a_t = (cp, pd, v)$. The possible *contact point* cp and the normal angle cn at the contact point are geometrically calculated from the initial 2D segmentation $\mathcal{S}_0$ as illustrated in Figure 6.

Monte-Carlo Sampling of push affordance: We generate M push affordances, $a_t^{[i]}$, $i \in 1..M$, from the possible points of contact points and contact normal by sampling a contact point and generating the $pd^{[i]} = cn^{[i]} + \delta; \delta \sim R(-5,5)$ (deg). The velocity v is fixed for all cases taking into account the quasi-static assumption.

N-step Information Gain: To make the framework more sample-efficient for real robot scenarios, we employ active action selection by formulating an N-step information gain criteria under the filtering setting based on the formulation of Eq.11. We recursively use the prediction step of the dual differentiable filter without the update step to compute the expected Information Gain for both model learning and object parameter inference for each sampled non-prehensile pushing action $\pi^{[i]} = a^i_{\tau_0:\tau_N}$ over $N-$step in future $\tau = \tau_0..\tau_N$

$$\begin{aligned} IG_N(\pi^{[i]}) \approx -\mathbb{E}_{p(\psi_{\tau_N},\phi_{\tau_N}|\pi^{[i]})}[ln(\overline{bel}^{[i]}(\psi_{\tau_N},\phi_{\tau_N}) \\ - ln(\overline{bel}^{[i]}(\psi_{\tau_0},\phi_{\tau_0})] \end{aligned} \tag{31}$$

where, $\overline{bel}^{[i]}(\psi_{\tau_N},\phi_{\tau_N})$ is the hypothetical predictive joint distribution after N-step by taking action $\pi^{[i]}$ without taking account the actual observation. For our case, the expectation is computed as KL-Divergence form for which the closed form solution exists for Multivariate Gaussian distributions [54].

$$\begin{aligned} IG_N(\pi^{[i]}) \approx D_{KL}[\mathcal{N}^{[i]}(\psi_{\tau_N},\phi_{\tau_N}|\overline{\mu}_{\tau_N},\overline{\Sigma}_{\tau_N})||\mathcal{N}^{[i]}(\psi_{\tau_0},\phi_{\tau_0}|\overline{\mu}_{\tau_0},\overline{\Sigma}_{\tau_0})] \\ \pi^* = argmax_{\pi^i} IG_N(\pi^{[i]}) \end{aligned} \tag{32}$$

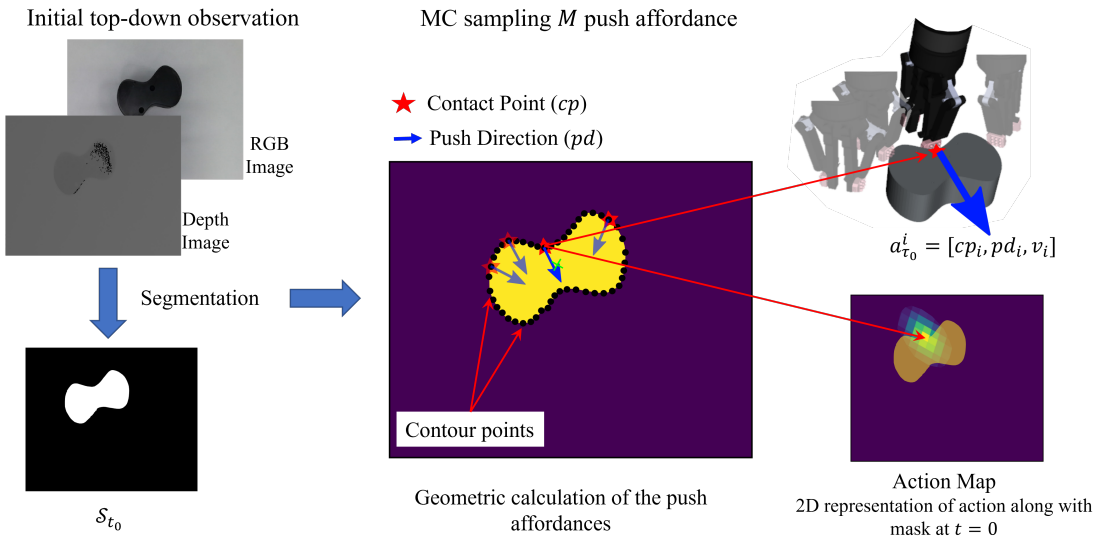

Figure 6: Visualization of Action Sampling from initial observation

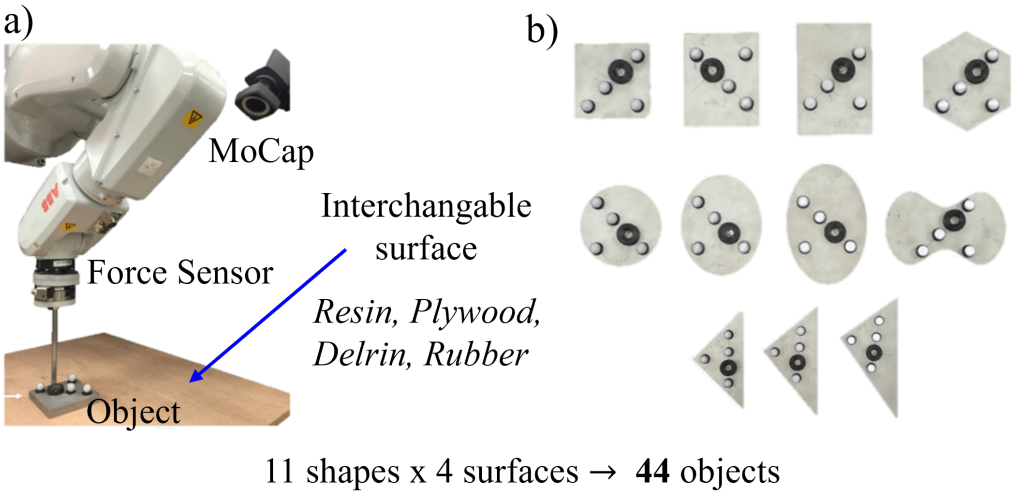

Figure 7: MIT Push Dataset Setup [55]. Part a) presents the data collection setup. Part b) presents the various objects in the dataset

4 Experiments

In this section, we explain the experiment setup and the results obtained from the proposed method, which is hereby referred to as VT-ADDF (Visuo-Tactile Active Dual Differentiable Filter). The closest state-of-the-art work to ours which dealt with the estimation of object parameters using robotic pushing was that of [29] and have taken it as the baseline. The baseline work utilised feature extraction using object pose, actions, and contact force information and a Multi-Output Regression Random Forest for data-driven regression modelling, which falls into non-predicitve approach. We re-implemented the baseline approach to the best of our capability and validated the published results on the MIT Push Dataset.

In addition, we performed extensive ablation studies 1) exploring the efficiency of the active approach vs random and uniform actions for learning and inference, 2) employing only vision for parameter estimation under the dual filtering setup (termed at V-DDF Visual-Dual Differentiable Filtering). For this, the *TacNet* was removed and the observations were reduced to only RGB-D. The rest of the framework and dual filtering setup with the active actions remained the same. 3) Study of dual filtering approach compared to joint filtering (termed as VT-JDF). In this, instead of performing separate parameter and pose updates, only a single UKF update equation was used [45].

4.1 Experimental Setups

We tested our approach and compared the baseline on 3 experimental setups.

4.1.1 Dataset - *MIT Push Dataset*

We utilized the MIT Push dataset, a state-of-the-art robotic pushing dataset [56] with 44 different objects, as shown in Figure 7. The dataset contains tactile, synthetic RGB-D, pose and parameter information. The objects were of 11 different shapes with varying mass, inertia, and 4 different surfaces - Abs, Delrin, Plywood, and Rubber-sheet were

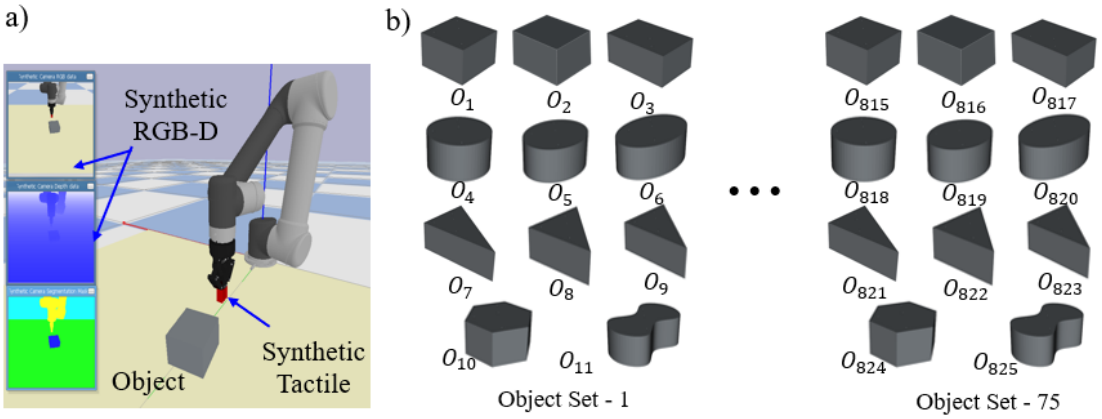

Figure 8: Simulation Setup-*Sim Robotac*. Part a) presents the PyBullet scene of the setup. Part b) presents the object set (825) objects used for the experiments.

Table 1: Parameter range for simulation setup

Property	**Range of values**
Mass (kg)	[0.2, 0.5, 0.8, 1.2]
μ	[0.35, 0.5, 0.7]
COM_x (m)	[-0.02, -0.015, 0, 0.015, 0.02]
COM_y (m)	[-0.02, -0.01, 0, 0.02, 0.03]
I_z (g.m2)	[0.5, 0.9, 1.15, 1.5]
Shapes	11 (Figure 8)

present. The center of mass of each object was slightly varied w.r.t. its geometric center. The dataset has almost 10,000 pushes for each object; however, we selected a partial subset of pushes with no acceleration and velocity of 30 mm/s in a total of 3750 pushes. As this was a pre-recorded dataset, we only present estimation with uniform actions rather than active actions. We selected this setup to validate the baseline results, as well as showcase that our proposed visuo-tactile dual differentiable filter can be utilized for different robotic environments.

4.1.2 Simulation Setup - *Sim Robotac*

We designed a simulation setup in PyBullet [57] to evaluate extensively our proposed approach as shown in Figure 8. In addition, it is possible to have a large set of objects with variations in physical parameters in the simulation, which is often difficult real robotic setup. The setup consisted of a simulated Robotiq gripper mounted on UR5 with a simulated tactile sensor attached to one of the finger pads. A synthetic RGB-D sensor was placed on top of the pushing area to simulate a visual sensor. In the simulation setup, 825 different objects were designed according to the parameters presented in Table 1. We used the simulation setup to perform extensive ablation studies, the results of which are presented in the following section.

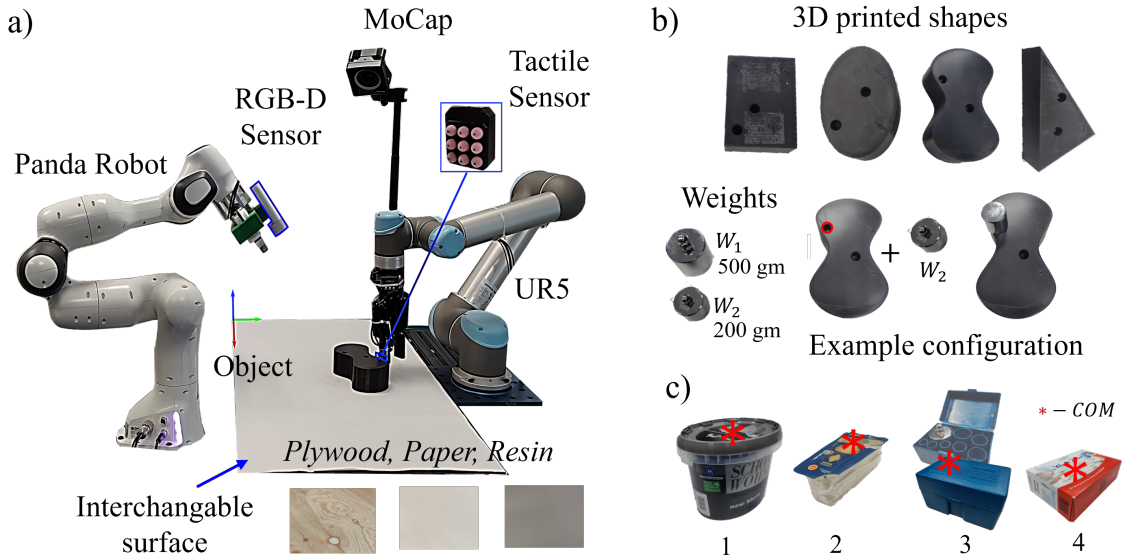

Figure 9: Robotic Setup - Real Robotac. Part a) presents the robotic setup. Part b) presents an overview of the configurable objects. Part c) presents the novel objects selected to test

4.1.3 Robotic Setup - *Real Robotac*

The robotic setup consists of Universal Robots (UR5) augmented with Robotiq two-finger Gripper and a Panda robotic manipulator as shown in Figure 9. The tactile sensor [58] is attached to the outer surface of the finger of the grippers on the Robotiq Gripper and an Azure DK RGB-D camera is rigidly attached to the Panda Gripper. The maximum allowed speed for the UR5 was 25 mm/s for safety constraints. The ground truth values of were collected using the motion capture system - Optitrack [59], whereas the ground truth values of the object parameters were computed from a CAD model of the objects. To obtain real objects with varying parameters, we designed configurable objects by 3D printing 4 shapes and adding additional weights at a precise location in the objects, changing their mass, center of mass, and inertia value. In addition, we utilized three different frictional surfaces, plywood, paper, and resin sheet, to vary the relative friction coefficient between the object and the pushing surfaces. In total, we had 48 different objects after all possible configurations. Furthermore, we have 4 novel daily objects as shown in Figure 9(c), which were not used in training and were kept only for testing. The objects had contrasting parameters (high mass of the paint box (Object 1), high friction of sugar cube box (Object 4) and the plywood surface, as well as shifted COM in cheese (Object 2) and weight box (Object 3).

4.2 Results

4.2.1 Learning - DDF Training

For training the networks used in the differentiable filter, we used a weighted combination of negative log-likelihood loss $\mathcal{L}_{\mathcal{NLL}}$ of the ground pose and parameter w.r.t. belief and mean squared error loss $\mathcal{L}_{\mathcal{MSE}}$ of contact forces and synthetic pose. Iterative training was performed using the Adam optimizer until the loss converged.

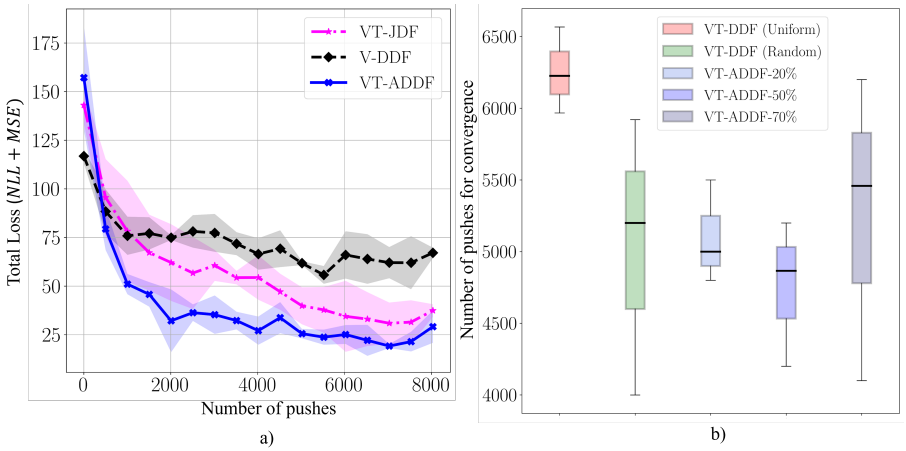

Figure 10: Learning results of the ablation studies in Sim-Robotac setup. Part (a) presents comparitive performance learning stability of VT-ADDF vs V-DDF vs VT-JDF. Part (b) presents the learning efficiency of different push action selection methods - Uniform, Active, Random

For the MIT Push data set, the time horizon was $t_H = 10s$ with a sampling rate of 10Hz. We divided the 3750 trajectories into 90% for training and 10% for inference. For the Sim-Robotac set-up, the time horizon was $t_H = 15s$ with a sampling rate of 10Hz. 90% of 825 objects were used for training and 10% for testing, cross-validated 5 times. We performed an ablation study using a uniform, random and active approach to take the action from the set of M-push affordances to train the filter. In addition, we also explored how much N- step lookahead is suitable. We chose N as 20% (= 3 secs), 50% (= 7.5 secs) and 70% (= 10.5) of the time horizon as future look-ahead steps for ablation study on active actions. We present the results of the ablation study in Figure 10 (b), for the efficiency of learning. In addition, we also present the validation loss plots to highlight the stability and learning performance of the proposed approach (VT-ADDF) compared to using only vision (V-DDF) and utilizing a joint differentiable filter (VT-JDF) in Figure 10 (a). For the Real-Robotac setup, the time horizon was $t_H = 10s$, with a sampling rate of 5 Hz and an active approach with 50% N step look-ahead (= 5 seconds) selected to train the dual differentiable filters. 90% of the 48 configurable 3D printed objects were used for training and 10% for testing.

4.2.2 Parameter Inference

For parameter inference of unknown (test) objects, we executed multiple push actions. At the end of each, the posterior belief of object parameters was utilised to initialize the belief for the next push. We present the results of the parameters m, μ, CoM_x, CoM_y, Iz inference for the ablation study in Sim-Robotac setup in Table. 2. For parameter inference, the N-step lookahead was selected as 50% of time horizon t_H for both Sim-Robotac and Real-Robotac setups. In addition, Figure 11 presents a closer look at the filtering action of the different ablation approaches and parameter inference convergence in the Sim-RoboTac setup. Further, the comparative parameter estimation results of the proposed approach (VT-ADDF) compared to the baseline work of [29] are presented in Table.3. We present separate estimation results for the novel objects which were not utilised for training, to evaluate the generalisation of the proposed approach compared to the baseline. As the range of values of different parameters like mass, friction co-efficient and inertia are different and the values often close to 0 (for the center of mass) we employ

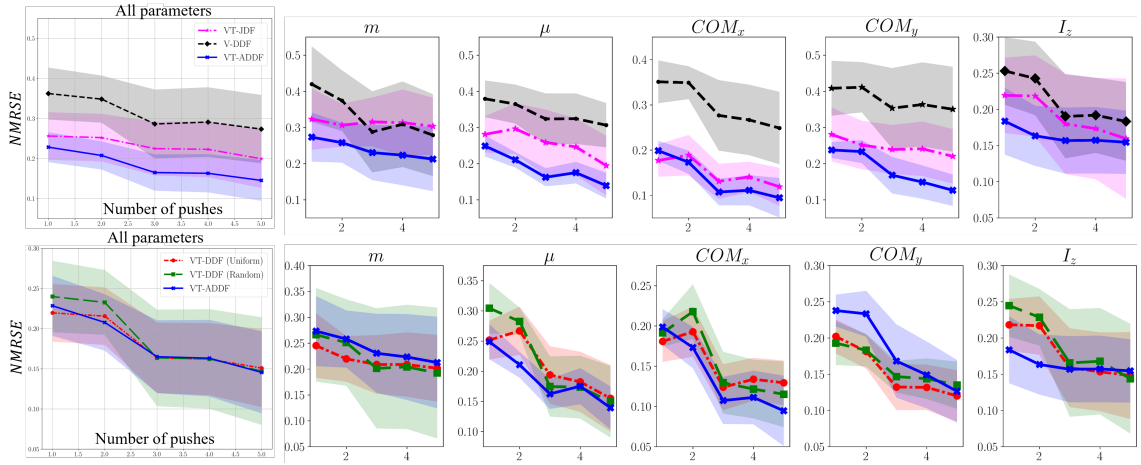

Figure 11: Inference result during the filtering step presented after each push action.

a normalised root mean square *NRMSE* [29] as metric to evaluate the performance over different parameters which is the root mean squared error divided by the range of values $(\psi_{max} - \psi_{min})$ of each parameters in the setups. Lower value, signifies better estimation.

Table 2: Inference result of *NRMSE* values of different parameters in the ablation study in Sim-Robotac setup

	mass	μ	com_x	com_y	I_z	Overall
VT-JDF	0.33 ± 0.08	0.19 ± 0.08	0.12 ± 0.04	0.22 ± 0.08	0.16 ± 0.08	0.20 ± 0.07
V-DDF	0.28±0.11	0.31±0.06	0.25±0.08	0.35±0.12	0.18±0.05	0.27±0.09
VT-ADDF	**0.21±0.09**	**0.14±0.04**	**0.09±0.04**	**0.13±0.04**	**0.15±0.03**	**0.14±0.05**
VT-DDF (Uniform)	0.20±0.06	0.16±0.06	0.13±0.03	0.12±0.03	0.12±0.06	0.15±0.05
VT-DDF (Random)	0.19±0.13	0.15±0.06	0.11±0.04	0.13±0.03	0.14±0.08	0.14±0.07

Table 3: Parameter Inference result of *NRMSE* value for proposed approach VT-ADDF compared to baseline work of [29] in various setups

	mass		*mu*		com_x		com_y		I_z	
Experimental Setup	Baseline	VT-ADDF	Baseline	VT-ADDF	Baseline	VT-ADDF	Baseline	VT-ADDF	Baseline	VT-ADDF
MIT Push Dataset	0.11±0.1	0.19±0.02	0.18±0.04	**0.17±0.02**	0.13±0.06	**0.10±0.04**	0.12±0.09	**0.09±0.07**	0.17±0.02	**0.16±0.01**
Sim Robotac	0.14±0.06	0.21±0.09	0.16±0.06	**0.14±0.04**	0.18±0.12	**0.09±0.04**	0.14±0.15	**0.13±0.04**	0.20±0.11	**0.15±0.03**
Real Robotac (RR)	0.25±0.12	**0.22±0.09**	0.14±0.03	0.19±0.06	0.12±0.08	**0.10±0.01**	0.20±0.1	**0.11±0.07**	0.16±0.05	**0.09±0.01**
RR Novel Objects	0.29±0.09	**0.20±0.05**	0.21±0.03	**0.18±0.06**	0.17±0.09	**0.11±0.05**	0.22±0.05	**0.12±0.05**	0.22±0.10	**0.15±0.08**

4.3 Discussion

In this work, we proposed a novel visuo-tactile-based active object parameter inference with a dual differentiable filter. The results of the learning ablation study show that active actions significantly improve the efficiency of the sample by around 20% (push actions) compared to uniform actions and have a lower variance than random action selection as presented in Figure 10(b). Moreover, it is shown that the dual filtering approach has more stable learning than the joint filter as presented in Figure 10(a). This demonstrates the efficacy of our proposed novel dual differentiable filtering approach compared to a joint differentiable filter method. Furthermore, our experimental results show that, by using only vision, the network fails to reduce the loss and tends to overfit. This is expected, as parameter estimation is difficult only via vision and has high error rates, leading to higher

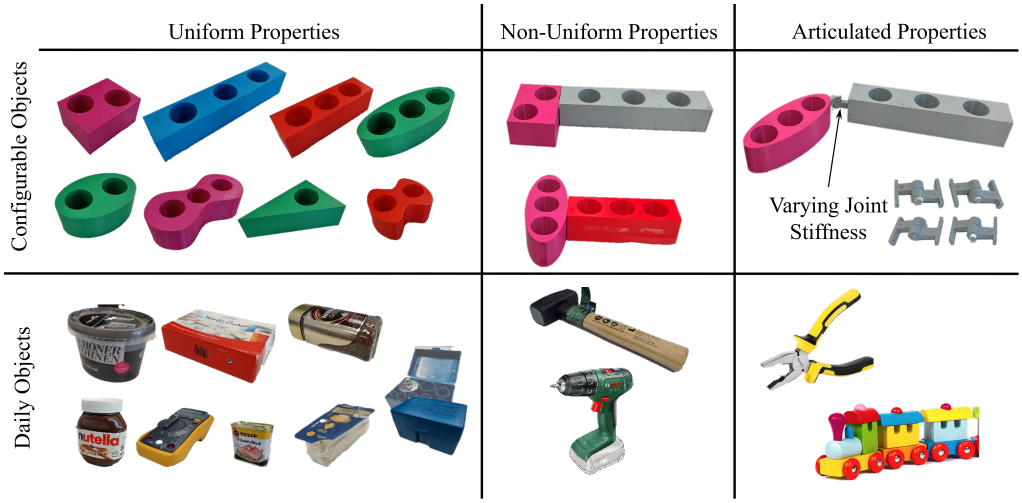

Figure 12: Extended range of objects with challenging parameter inference.

loss values.

The results obtained from the parameter inference show that our proposed approach performs consistently on different experimental setups, MIT Push Dataset, Sim-Robotac and Real-Robotac, compared to baseline work of [29], which fails to generalize for novel objects in real robotic setup. Moreover, the limitation of providing ground-truth pose information in the baseline approach is elevated by our proposed ADDF approach during the inference step. Furthermore, the ablation study shows that active actions have a better estimation of parameters compared to uniform and with lower variance than random actions with the same number of push actions. Compared to using only vision, the visuo-tactile dual differentiable approach performs much better, especially in parameters like the center of mass prediction, as well as being more stable and accurate than the joint filtering approach. Through the different setups, we also show that the proposed visual-tactile-based dual differentiable filter for parameter inference is agnostic to robotic setups as long as sufficient visual and tactile information is present.

From the limitations, our proposed ADDF requires separate training of *VisNet* with *MSE* loss to obtain pose information for novel objects which are visually quite different from the training set. Instead of using RGB-D as visual observations and a 2D pose estimation network, it will be viable to use point clouds and avoid the requirement of using pose estimation altogether or use recent one-shot pose estimation approaches. In addition, it will be interesting to avoid the requirement of having ground truth states and parameter values during training, as well as to develop a framework which can discover physical object representations.

5 Future Work and Improvement

In the above approach, we considered the state estimation of rigid objects with uniform physical properties. However, various daily objects can have non-uniform properties (e.g. hammer) or can be articulated (e.g. doors, drawers). We propose an extension of the method to estimate the parameters of a more diverse range of objects, uniform, non-uniform, and articulated, as shown in Figure 12.

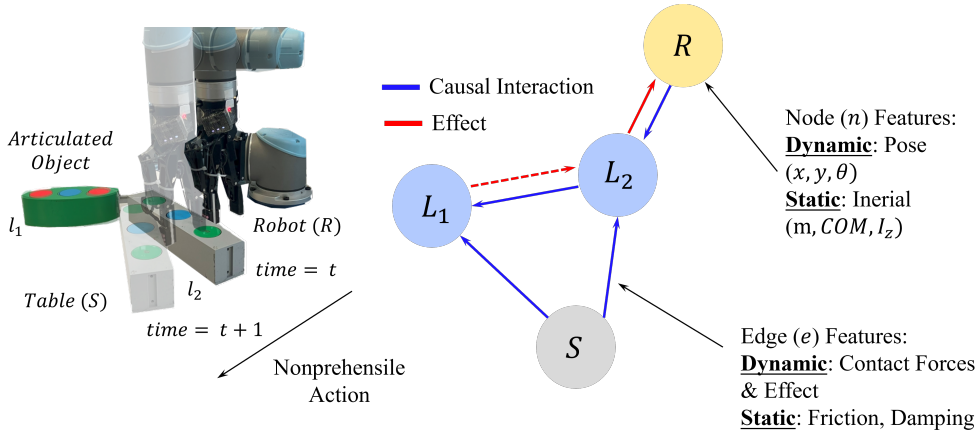

Figure 13: Graph generation of an illustrative articulated object with 2 links. The edges has capability to discriminate between cause and effect depending how the robot is interacting with the object. Node captures the states, whereas the edges captures interaction forces

.

In order to handle such diverse objects, we propose to use Graph Networks [60] which provides a strong inductive bias to capture the complex interaction dynamics, which is difficult to capture by standard feedforward networks. In addition, as the state-space dimension increases, it becomes difficult for feed-forward networks to generalize well. In contrast, Graph Networks utilize the same structure and the fully connected network to predict states through the multiple message transmission mechanism [60]. Figure 13 illustrates how the interaction between the robot and an articulated object is represented by a graph with nodes and features. Furthermore, Figure 14 shows how multi-messaging is used to update the node and edge features. The edge effect (marked by a red arrow) that returns to the robot is utilized for tactile prediction. The initial results show that such a representation is quite efficient in generalizing compared to the previous approach.

Finally, with the addition of Graph Networks, improved shape perception and the addition of 3D projective transformations [61], the updated framework of Active visuo-tactile rigid object property inference is presented in Figure 15 which is inspired from human like 'predictive' top-down processing.

This is ongoing work as of 5^{th} December 2023 (deliverable submission) and will be submitted to the **Transactions in Robotics (TRO)** by January 15^{th} 2024.

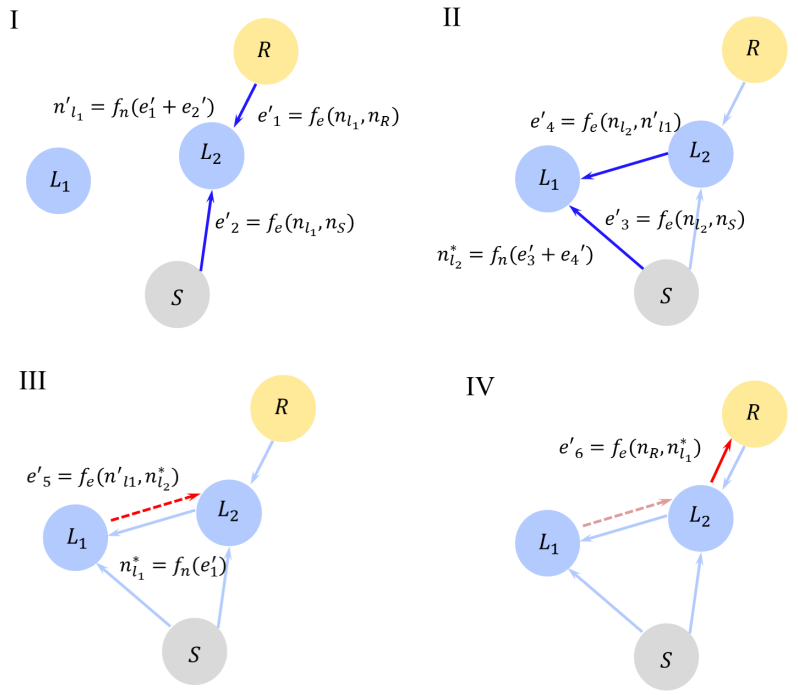

Figure 14: Graph Propagation to predict the next step state (process model). It utlises multiple message passing (I-IV) unlike single pass of feed-forward networks. The node and edge functions f_e, f_n are approximated by fully connected networks

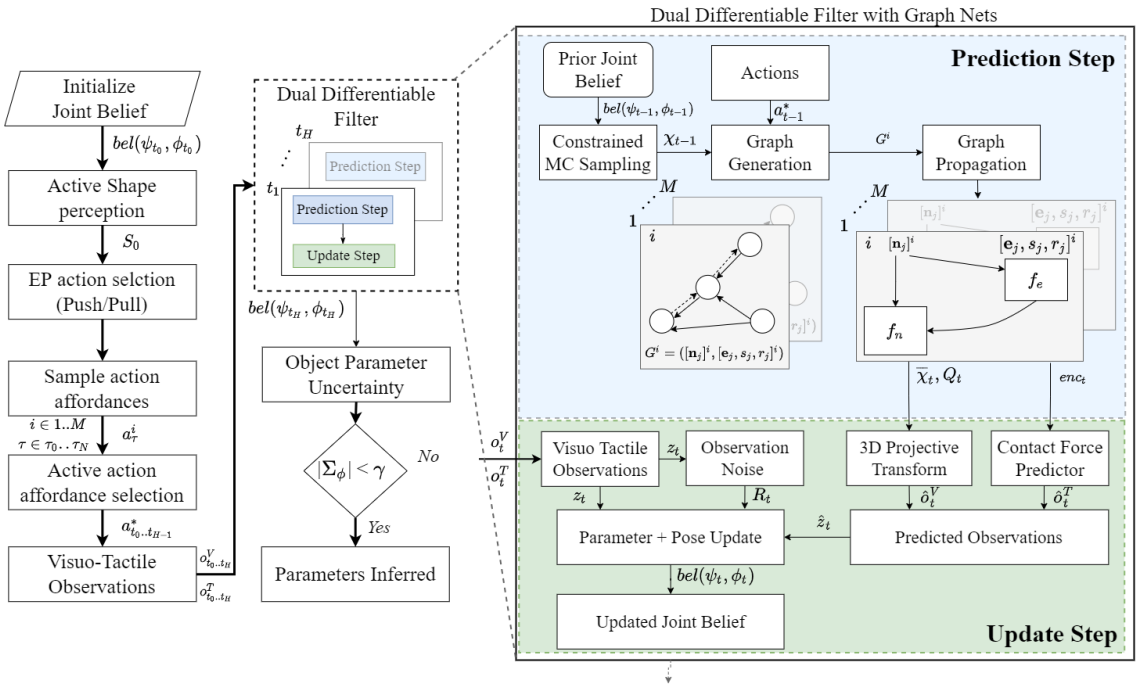

Figure 15: Proposed improved framework on Active Differentiable Filter approach for object property inference

6 Conclusion

In this research project, part of INTUITIVE, we investigated the combination of vision, tactile, and proprioception for robotic manipulators. We proposed a novel active dual dif-

ferentiable filtering approach to address the problem, which draws inspiration from the predictive processing abilities found in humans. To demonstrate the capability of the proposed approach, we selected the problem of estimating the properties of rigid objects via non-prehensile pushing. The filter represents the object state systematically into time-varying and time-invariant factors and takes into account the current state of the object, action (proprioception) of the robot to predict the next state of the object as well as expected visual and tactile observation. The proposed differentiable filter first learns an object interaction model using known objects, which is then used to infer novel objects under differentiable filter settings. We present a novel formulation of active action selection with the differentiable filter as one of the key contributions. The generalizability of the framework makes it suitable for real robotic applications and opens the possibility of exploring the approach for other interaction techniques for object parameter estimation such as grasping, prehensile pulling, etc. In addition, we presented ongoing work using Graph Networks as inductive bias within the differentiable filtering setting, which provides capability to be used as a general purpose perception framework for downstream manipulation tasks.

Bibliography

[1] C. H. An and J. M. Hollerbach, "The role of dynamic models in cartesian force control of manipulators," *The International Journal of Robotics Research*, vol. 8, no. 4, pp. 51–72, 1989.

[2] N. Mavrakis and R. Stolkin, "Estimation and exploitation of objects' inertial parameters in robotic grasping and manipulation: A survey," *Robotics and Autonomous Systems*, vol. 124, p. 103374, 2020.

[3] Y. Li, A. Sena, Z. Wang, X. Xing, J. Babic, E. H. van Asseldonk, and E. Burdet, "A review on interaction control for contact robots through intent detection," *Progress in Biomedical Engineering*, 2022.

[4] Y. Li, G. Ganesh, N. Jarrassé, S. Haddadin, A. Albu-Schaeffer, and E. Burdet, "Force, impedance, and trajectory learning for contact tooling and haptic identification," *IEEE Transactions on Robotics*, vol. 34, no. 5, pp. 1170–1182, 2018.

[5] J. Bohg, K. Hausman, B. Sankaran, O. Brock, D. Kragic, S. Schaal, and G. S. Sukhatme, "Interactive perception: Leveraging action in perception and perception in action," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1273–1291, 2017.

[6] P. Allen, "Surface descriptions from vision and touch," in *Proceedings. 1984 IEEE International Conference on Robotics and Automation*. Institute of Electrical and Electronics Engineers. [Online]. Available: https://doi.org/10.1109/robot.1984.1087191

[7] P. K. Allen, "Integrating vision and touch for object recognition tasks," *The International Journal of Robotics Research*, vol. 7, no. 6, pp. 15–33, Dec. 1988. [Online]. Available: https://doi.org/10.1177/027836498800700603

[8] A. Kimoto and Y. Matsue, "A new multifunctional tactile sensor for detection of material hardness," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 4, pp. 1334–1339, 2011.

[9] L. Seminara, P. Gastaldo, S. J. Watt, K. F. Valyear, F. Zuher, and F. Mastrogiovanni, "Active haptic perception in robots: A review," *Frontiers in Neurorobotics*, vol. 13, Jul. 2019. [Online]. Available: https://doi.org/10.3389/fnbot.2019.00053

[10] E. Burdet *et al.*, *Human Robotics: neuromechanics and motor control*. MIT press, 2013.

[11] M. Kaboli *et al.*, "Tactile-based manipulation of deformable objects with dynamic center of mass," in *ICHR*. IEEE, 2016.

[12] R. Bajcsy, Y. Aloimonos, and J. Tsotsos, "Revisiting active perception," *Autonomous Robots*, vol. 42, 02 2018.

[13] L. Seminara *et al.*, "Active haptic perception in robots: a review," *Front. in Neurorobotics*, vol. 13, p. 53, 2019.

[14] Q. Li *et al.*, "A review of tactile information: Perception and action through touch," *IEEE Trans. on Rob.*, vol. 36, no. 6, pp. 1619–1634, 2020.

[15] K. Friston and S. Kiebel, "Predictive coding under the free-energy principle," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 364, no. 1521, pp. 1211–1221, May 2009. [Online]. Available: https://doi.org/10.1098/rstb.2008.0300

[16] K. Friston, "The free-energy principle: a unified brain theory?" *Nature Reviews Neuroscience*, vol. 11, no. 2, pp. 127–138, Jan. 2010. [Online]. Available: https://doi.org/10.1038/nrn2787

[17] Z. Xu, J. Wu, A. Zeng, J. B. Tenenbaum, and S. Song, "Densephysnet: Learning dense physical object representations via multi-step dynamic interactions," *arXiv preprint arXiv:1906.03853*, 2019.

[18] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Estimation of inertial parameters of manipulator loads and links," *The International Journal of Robotics Research*, vol. 5, no. 3, pp. 101–119, 1986.

[19] C. Wang, X. Zang, X. Zhang, Y. Liu, and J. Zhao, "Parameter estimation and object gripping based on fingertip force/torque sensors," *Measurement*, vol. 179, p. 109479, 2021.

[20] Y. Yu, T. Arima, and S. Tsujio, "Estimation of object inertia parameters on robot pushing operation," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. IEEE, 2005, pp. 1657–1662.

[21] Y. Yu, K. Fukuda, and S. Tsujio, "Estimation of mass and center of mass of graspless and shape-unknown object," in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, vol. 4. IEEE, 1999, pp. 2893–2898.

[22] Z. Zhao, X. Li, C. Lu, and Y. Wang, "Center of mass and friction coefficient exploration of unknown object for a robotic grasping manipulation," in *2018 IEEE International Conference on Mechatronics and Automation (ICMA)*, 2018, pp. 2352–2357.

[23] S. Tanaka, T. Tanigawa, Y. Abe, M. Uejo, and H. T. Tanaka, "Active mass estimation with haptic vision," in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, vol. 3. IEEE, 2004, pp. 256–261.

[24] K. Yao, M. Kaboli, and G. Cheng, "Tactile-based object center of mass exploration and discrimination," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, 2017, pp. 876–881.

[25] B. Sundaralingam and T. Hermans, "In-hand object-dynamics inference using tactile fingertips," *IEEE Transactions on Robotics*, vol. 37, no. 4, pp. 1115–1126, 2021.

[26] J. Wu, I. Yildirim, J. J. Lim, B. Freeman, and J. Tenenbaum, "Galileo: Perceiving physical object properties by integrating a physics engine with deep learning," *Advances in neural information processing systems*, vol. 28, 2015.

[27] C. Song and A. Boularias, "A probabilistic model for planar sliding of objects with unknown material properties: Identification and robust planning," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5311–5318.

[28] ——, "Learning to slide unknown objects with differentiable physics simulations," in *Robotics science and systems*, 2020.

[29] N. Mavrakis, R. Stolkin *et al.*, "Estimating an object's inertial parameters by robotic pushing: a data-driven approach," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 9537–9544.

[30] M. Kaboli, A. Long, and G. Cheng, "Humanoids learn touch modalities identification via multi-modal robotic skin and robust tactile descriptors," *Advanced Robotics*, vol. 29, no. 21, pp. 1411–1425, 2015.

[31] M. Kaboli *et al.*, "In-hand object recognition via texture properties with robotic hands, artificial skin, and novel tactile descriptors," in *ICHR*. IEEE, 2015, pp. 1155–1160.

[32] A. Kloss, M. Bauza, J. Wu, J. B. Tenenbaum, A. Rodriguez, and J. Bohg, "Accurate vision-based manipulation through contact reasoning," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 6738–6744.

[33] P. K. Murali, A. Dutta, M. Gentner, E. Burdet, R. Dahiya, and M. Kaboli, "Active visuo-tactile interactive robotic perception for accurate object pose estimation in dense clutter," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4686–4693, 2022.

[34] M. A. Lee, Y. Zhu, K. Srinivasan, P. Shah, S. Savarese, L. Fei-Fei, A. Garg, and J. Bohg, "Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8943–8950.

[35] J. Van der Kamp, R. Oudejans, and G. Savelsbergh, "The development and learning of the visual control of movement: An ecological perspective," *Infant Behavior and Development*, vol. 26, no. 4, pp. 495–515, 2003.

[36] Z. Zhang *et al.*, "Beyond point clouds: Fisher information field for active visual localization," in *IEEE ICRA*, 2019.

[37] M. T. Mason, "Mechanics and planning of manipulator pushing operations," *The International Journal of Robotics Research*, vol. 5, no. 3, pp. 53–71, 1986.

[38] T. P. Minka, "A family of algorithms for approximate bayesian inference," Ph.D. dissertation, Massachusetts Institute of Technology, 2001.

[39] M. Baltieri and T. Isomura, "Kalman filters as the steady-state solution of gradient descent on variational free energy," *arXiv preprint arXiv:2111.10530*, 2021.

[40] G. Oliver, P. Lanillos, and G. Cheng, "An empirical study of active inference on a humanoid robot," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 14, no. 2, pp. 462–471, Jun. 2022. [Online]. Available: https://doi.org/10.1109/tcds.2021.3049907

[41] R. Jonschkowski, D. Rastogi, and O. Brock, "Differentiable particle filters: End-to-end learning with algorithmic priors," *arXiv preprint arXiv:1805.11122*, 2018.

[42] K.-T. Yu and A. Rodriguez, "Realtime state estimation with tactile and visual sensing. application to planar manipulation," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 7778–7785.

[43] A. S. Lambert, M. Mukadam, B. Sundaralingam, N. Ratliff, B. Boots, and D. Fox, "Joint inference of kinematic and force trajectories with visuo-tactile sensing," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3165–3171.

[44] T. Haarnoja, A. Ajay, S. Levine, and P. Abbeel, "Backprop kf: Learning discriminative deterministic state estimators," *Advances in neural information processing systems*, vol. 29, 2016.

[45] A. Kloss, G. Martius, and J. Bohg, "How to train your differentiable filter," *Autonomous Robots*, vol. 45, no. 4, pp. 561–578, 2021.

[46] A. Dutta, E. Burdet, and M. Kaboli, "Push to know!–visuo-tactile based active object parameter inference with dual differentiable filtering," *arXiv preprint arXiv:2308.01001*, 2023.

[47] S. Thrun, "Probabilistic robotics," *Communications of the ACM*, vol. 45, no. 3, pp. 52–57, 2002.

[48] D. Ebeigbe, T. Berry, M. M. Norton, A. J. Whalen, D. Simon, T. Sauer, and S. J. Schiff, "A generalized unscented transformation for probability distributions," *ArXiv*, 2021.

[49] M. Wüthrich, C. G. Cifuentes, S. Trimpe, F. Meier, J. Bohg, J. Issac, and S. Schaal, "Robust gaussian filtering using a pseudo measurement," in *2016 American Control Conference (ACC)*. IEEE, 2016, pp. 3606–3613.

[50] G. Ziyan, A. Elibol, and N. Y. Chong, "Planar pushing of unknown objects using a large-scale simulation dataset and few-shot learning," in *2021 IEEE 17th international conference on automation science and engineering (CASE)*. IEEE, 2021, pp. 341–347.

[51] J. Liu and M. West, "Combined parameter and state estimation in simulation-based filtering," in *Sequential Monte Carlo methods in practice*. Springer, 2001, pp. 197–223.

[52] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[53] C. B. Do, "The multivariate gaussian distribution," *Section Notes, Lecture on Machine Learning, CS*, vol. 229, 2008.

[54] J. Duchi, "Derivations for linear algebra and optimization," *Berkeley, California*, vol. 3, no. 1, pp. 2325–5870, 2007.

[55] "The mcube lab - push dataset," https://mcube.mit.edu/push-dataset/index.html, (Accessed on 03/02/2023).

[56] K. Yu *et al.*, "More than a million ways to be pushed. a high-fidelity experimental dataset of planar pushing," in *IEEE IROS*. IEEE, 2016, pp. 30–37.

[57] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning," 2016.

[58] Contactile, "Contactile," https://contactile.com/, 2022, [Online; accessed 15092022].

[59] "Optitrack - motion capture systems," https://optitrack.com/, (Accessed on 03/02/2023).

[60] A. Sanchez-Gonzalez, N. Heess, J. T. Springenberg, J. Merel, M. Riedmiller, R. Hadsell, and P. Battaglia, "Graph networks as learnable physics engines for inference and control," in *International Conference on Machine Learning*. PMLR, 2018, pp. 4470–4479.

[61] R. Hartley and A. Zisserman, *Projective Geometry and Transformations of 3D*, 2nd ed. Cambridge University Press, 2004, p. 65–86.