Interactive Shared Visuo-Tactile Perception and Learning in Robotics

Prajval Kumar Murali

Submitted in fulfilment of the requirements for the
Degree of Doctor of Philosophy

School of Engineering
College of Science and Engineering
University of Glasgow

September 2024

Abstract

Humans possess the ability to seamlessly integrate perceptual information from vision and tactile sensing to maintain a high-level cognitive understanding of the environment. Similarly, leveraging vision and tactile sensing can enable robots to interact with novel objects in unstructured environments. This thesis presents novel approaches for visuo-tactile perception and learning in robotics, focussing on object pose estimation, recognition, and reconstruction.

For robust pose estimation of unknown objects in dense clutter, a novel recursive filtering formulation termed translation-invariant Quaternion filter (TIQF) and its global-optimal version stochastic TIQF (S-TIQF) with visuo-tactile point cloud data are proposed in this thesis. A two-robot team with vision and tactile sensing autonomously declutters the scene and retrieves the pose of the target object which can be opaque or transparent using S-TIQF. Moreover, S-TIQF is deployed to correct hand-eye calibration with arbitrary objects in-situ which is necessary for shared perception. In addition to rigid objects, the pose tracking of articulated objects is a challenging task that requires integration of vision and tactile sensing. A novel Manifold Unscented Kalman Filter method on the SE(3) Lie Group termed ArtReg is presented, which is used for tracking the pose of objects. Using ArtReg, a novel framework is designed for detecting, tracking, and the goal-driven manipulation of unknown objects (single, multiple, or articulated) without assuming any prior knowledge regarding object shape or dynamics.

This thesis also presents a new vision-to-tactile cross-modal learning approach for object recognition where the network is trained with dense visual point clouds and tested with sparse point clouds acquired from tactile sensors. A novel unsupervised domain adaptation loss function is proposed to minimise the gap between the visual and tactile domains. Cross-modal adaptation allows the robotic system to switch to tactile sensing in case vision sensing is compromised, thereby increasing the robustness of the system.

Object reconstruction is another fundamental perceptual challenge that enables downstream tasks such as pose estimation and recognition. This thesis introduces a novel deep learning-based 3D object reconstruction approach utilising sparse tactile point cloud data to accurately recover the geometry of category-level unknown transparent objects leveraging only synthetic data for training. Furthermore, it is also demonstrated with visuo-tactile point clouds for opaque objects, wherein the tactile data are used to refine the shape in regions of uncertainty in the visual data.

The proposed methods have been rigorously validated with extensive experiments on standard datasets and robot experiments and have been demonstrated to outperform state-of-the-art approaches.

This thesis is dedicated to my father.

Acknowledgement

Firstly, I would like to thank the BMW Group for generously supporting the doctoral research through the ProMotion programme. The entire Ph.D. work was funded by the BMW Group, with all phases of my Ph.D. work: design, implementation, evaluation, and writing articles conducted at the BMW Group. I would like to wholeheartedly thank my supervisors, Prof. Mohsen Kaboli at the BMW Group and Prof. Bernd Porr at the University of Glasgow for their guidance and support during the doctoral study. Prof. Kaboli guided me to think deeply about scientific problems to identify research gaps and develop novel ideas. Prof. Porr provided invaluable expert guidance and profound insights that challenged and inspired me to think outside the box about my research. I would like to thank both my supervisors for the encouragement, expertise, and guidance without which this thesis would not have been possible. I would like to thank Prof. Ravinder Dahiya for providing guidance and feedback to my Ph.D. I also thank the BMW Group ProMotion programme management for providing various soft-skill training opportunities within the ProMotion programme that have helped to hone my skills.

My sincere appreciation to the RoboTac Lab at the BMW Group for providing a stimulating and supportive environment to conduct research. I would like to thank Anivan, my fellow PhD colleague and close friend. The brainstorming sessions, late night experiments to meet conference deadlines, and coffee sessions are something that I cherish. I would like to thank Michael for the intellectual camaraderie and discussions. I would like to thank the former colleagues at BMW: Xavi, Cong, Chiara, Aitana. The discussions, collaborations, and friendships that I have formed with my colleagues have made my Ph.D. journey a truly rewarding experience.

I also thank my former mentors, Dr. Daniele Pucci, Dr. Silvio Traversaro, and Prof. Fulvio Mastrogiovanni. Their teaching and guidance provided a solid foundation for conducting my doctoral research. A special mention goes to my friends Haresh, Prashant, and Yeshi and my friends back home in Bangalore, Aniruddha, Akshar, Abhishek, Dilip, Prateek, Vasujith, Srivatsav and the rest for providing much needed moments of respite.

Most importantly, I would like to thank my family for their unwavering support and

encouragement. This thesis is dedicated to my father, Murali Rajaram, who cherished every achievement of mine, who sacrificed so much so that I could pursue my own passions and who always encouraged me to dream big and not get complacent in life, but very sadly, who is not here to witness this moment. This and every other achievement in my life is for you, daddy. To my mother, Asha Murali, for your unwavering love and motivation, patiently listening to me always and providing strength through the challenging periods. To my grandfather, whose positive spirit and outlook towards life inspires us all. To my late grandmother, for taking care of me during my formative years. To my aunt, uncle, and cousin sister who understands the Ph.D. journey herself and always providing words of reassurance and the importance of faith. To our sweet little dog, Knobby, who has helped me in more ways than he realises. To my life partner, best friend, and loving wife, Archana, your patience, love, and emotional support have been my anchor throughout this journey. I am deeply indebted to my family without whom this thesis would not have been possible.

Thank You
Prajval Kumar Murali

Contents

List of Tables

List of Figures

Declaration

I declare that, except where explicit reference is made to the contribution of others, that this dissertation is the result of my own work and has not been submitted for any other degree at the University of Glasgow or any other institution.

Prajval Kumar Murali September 2024

Part I

Background and Thesis Context

Chapter 1

Introduction

1.1 Motivation

Among the various sensory modalities present in the human body, vision and tactile sensing are primarily used for perceiving and interacting with various objects within our environment. The human brain seamlessly integrates data from various sensing modalities, enabling us to interact with our surroundings (Hillis et al., 2002). In fact, visual and tactile perception are known to be integrated in a statistically optimal manner in the brain (Ernst and Banks, 2002). Robots should also be able to achieve a similar level of scene understanding, given that they are similarly equipped, for example, with visual and tactile sensing. Visual perception offers comprehensive scene information, including colour, brightness, shapes, and the position/ orientation (pose) of objects around us. In contrast, tactile sensing delivers detailed local and complementary information such as texture, hardness, mass, temperature, and so on (Liu et al., 2020). Visual sensing is susceptible to environmental factors, including low light conditions and the transparency or specularity of objects. Conversely, tactile sensing remains unaffected by these limitations. The shared perception between complementary visual and tactile sensing modalities offers a comprehensive and accurate scene representation, as well as addressing the weaknesses inherent in individual sensor systems (Murali et al., 2022e). Similarly to humans, robots possess the capability to augment their perceptual data through deliberate manipulative actions, a technique referred to as interactive perception, which cultivates a symbiotic interplay between action and perception. (Bohg et al., 2017). Thus, by capitalising on shared and interactive perception, robots can potentially enhance their autonomy and operational efficacy in real-world scenarios. While there are many applications wherein visuo-tactile perception can be leveraged for robotics, this thesis focusses on foundational perceptual applications such as object pose estimation, reconstruction, and recognition through shared and interactive

visuo-tactile perception.

A fundamental challenge inherent in any robotic perceptual system is the pose estimation task which is to find the position and orientation of an arbitrary object (with known or unknown object model). Pose estimation facilitates subsequent tasks, such as autonomous object interaction involving manipulation actions (e.g., pick-and-place or push to target pose). Consequently, vision-based pose estimation has undergone extensive research by the computer vision community and has been implemented in a myriad of industrial applications (Zou et al., 2023). Vision-based pose estimation techniques are sensitive to ambient lighting conditions, occlusions in unstructured settings, and object surface properties (such as transparency or specularity) (Luo et al., 2017). Tactile sensing is robust to such environmental and object properties and can enhance and improve the vision-based pose estimate (Li et al., 2020a). Similarly to humans, tactile sensing can be used to verify and correct residual uncertainties inherent in vision-based sensing (Strub et al., 2014). Furthermore, objects may lie in dense clutter in unstructured scenarios. Since tactile sensing is an inherent active perceptual system in which an interaction with objects is needed to extract sensory information, it can be leveraged to perform manipulation actions to rearrange the scene to improve overall perception (Bohg et al., 2017). Apart from dense clutter scenarios, there are many types of complex articulated objects in our environments which have inherent articulated kinematic chains such as drawers, reading glasses, washing machines, and so on. Tactile sensing is crucial for robots to interact safely with unknown articulated objects, to determine the joint limits during manipulation (Martín-Martín and Brock, 2022). Hence, the real-time pose estimation and tracking of such articulated objects combining visuo-tactile perception is crucial for the safe robot interaction.

Adjacent to the object pose estimation task, object recognition is another fundamental perception task for robots. Similarly to pose estimation techniques, visual and tactile sensing data can be fused to extract information about the object (Liu et al., 2017). Although multi-modal fusion helps augment the information from complementary sensing modalities, cross-modal perception can benefit the robot by relying on the other sensing modality if one modality is unavailable or erroneous (Falco et al., 2019). A motivating example is when we enter a dark room where the lights are turned off, we are able to locate a previously seen object based on exploring with our hands (with tactile perception) only, even if we have not previously touched the object before. Empirical research has documented instances of cross-modal perception in human infants (Streri and Gentaz, 2004). In visuo-tactile cross-modal perception, the robot adapts the prior knowledge gained from one modality (for instance, vision), the so-called source domain, with the second modality (for instance, tactile) or target domain, in order to complete the task with the second modal-

ity (Murali et al., 2022e). This avoids the additional expensive overhead of data collection and annotation with the target domain data. It also improves the resilience of the robotic system against possible sensor failures and maintains the same level of functionality while operating in unstructured settings.

Another significant perceptual challenge addressed in this dissertation is the three-dimensional (3D) reconstruction of objects at the category level when their specific instances are not previously known. For opaque objects, a precise 3D scanner can be employed to generate an accurate 3D representation of the object by systematically manoeuvring the scanner around the target entity (Han et al., 2019). However, for objects with non-Lambertian surfaces such as transparent and specular objects, these scanners that rely on typical depth-sensing mechanisms (such as structured light or time-of-flight cameras) provide incomplete reconstructions (Ihrke et al., 2010). Tactile sensing can be used independently or in conjunction with visual sensing to reconstruct such objects by probing or moving the tactile sensor over the surface of the objects in an intelligent manner (Jiang et al., 2023). The sparsity of tactile data necessitates the development of new methodologies that learn to reconstruct shapes based on prior knowledge of similar objects. Furthermore, tactile perception can serve to validate and refine the reconstructed 3D geometries by exploring regions of ambiguity derived from the preliminary visual reconstruction.

It is evident that visuo-tactile perception can significantly enhance the ability of robotic systems to interact with the environment in a safe and robust manner. However, numerous challenges associated with multi-modal perception will be elucidated in the following section.

1.2 Challenges

The main challenges for the integration of visuo-tactile perception is due to the weak pairing between visual and tactile data: (a) variation in the density of information from each modality, (b) scale gap as vision sensors can capture the global scene while tactile sensors capture local object geometry, (c) temporal misalignment as vision sensors capture data in one shot while tactile sensors capture data sequentially, and (d) tactile data are inherently action conditioned as data depends on the type of action that is performed (Liu et al., 2020). The visual and tactile modality also provides different types of raw data, for example, images from cameras and force or pressure values from tactile sensors (Dahiya, 2019). The data must be transformed into a uniform domain, either at the raw data level or at the feature extraction level, to facilitate effective integration (Navarro-Guerrero et al., 2023).

Considering the problem of visuo-tactile based object pose estimation, the difference

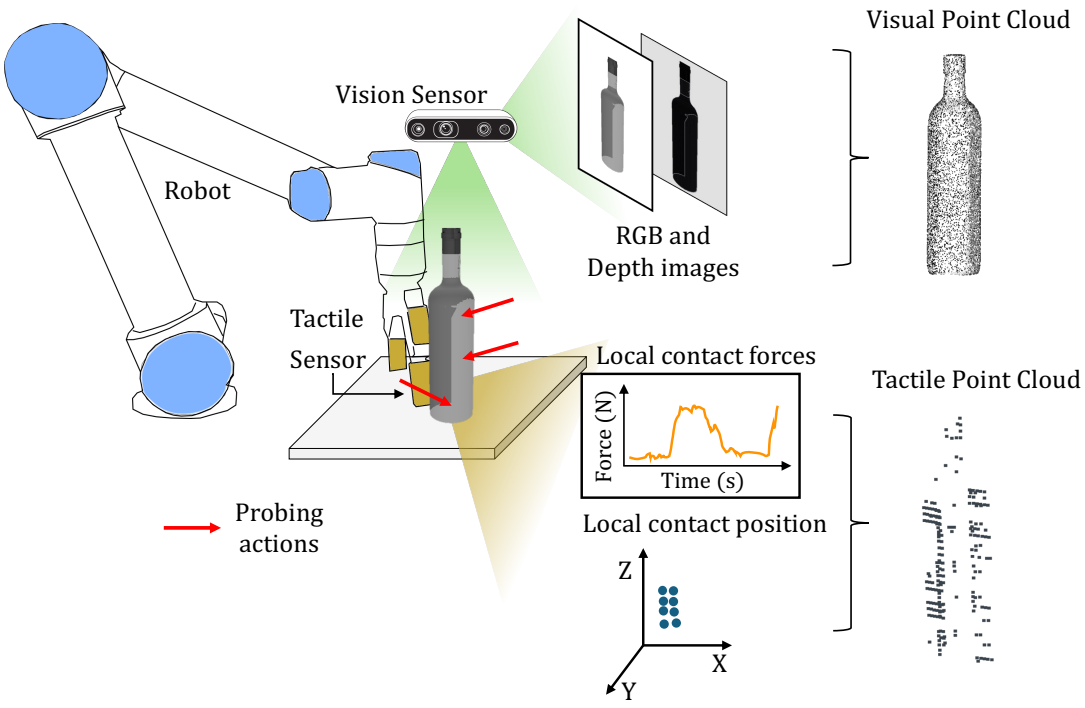


Figure 1.1: Schematic description of vision and tactile data of an arbitrary object extracted by a robot. Visual data include RGB and depth images which can be converted to dense 3D point clouds. In contrast, tactile sensing provides local contact force information and 3D location of the point of contacts. By probing many regions of the object with a robot, a sparse point cloud consisting of 3D tactile contact locations can be extracted.

in density of information from visual and tactile data poses a significant challenge (Bimbo et al., 2015). A popular method for 3D pose estimation is through point cloud registration (Huang et al., 2021b). Point clouds also offer a convenient data structure for mapping the visual and tactile data into a common domain as seen in Fig. 1.1. Common RGB-D cameras can capture a wide field-of-view of the scene in one shot providing dense point clouds (typically $10^4 - 10^6$ points) (Horaud et al., 2016). In contrast, tactile data are extracted only when objects are contacted by the sensor, and the robot needs to contact the entire surface of the object through multiple repetitive probing actions. Based on the contact forces, the 3D contact point can be derived from the kinematics of the robot and collected in a tactile point cloud. The tactile point clouds are typically sparse ($\sim 10^2$ points) and time consuming for data collection (Murali et al., 2022b). Hence, the sparse-dense correspondence estimation between the point clouds poses a technical challenge (Bimbo et al., 2015). The majority of methodologies in point cloud registration, whether geometric-based or learning-based, initially identify putative correspondences between point clouds using feature extraction

techniques (Huang et al., 2021b). These feature extraction methods assume sufficiently dense point clouds as the nearest neighbours of query points are used to extract the feature information. Moreover, techniques such as the Iterative Closest Point (ICP) (Besl and McKay, 1992) and similar methods perform simultaneous correspondences estimation and pose registration in an iterative manner (Pomerleau et al., 2013). But these techniques are typically sensitive to local-minima and need to be properly initialised (Pomerleau et al., 2013). In addition, tactile point clouds are extracted sequentially based on the actions performed, whereas visual point clouds are extracted in a single-shot manner. The object may also move while performing tactile probing actions, and the visuo-tactile sensors have to be in closed-loop with the interactive manipulation actions in order to track the objects. This challenge is further exemplified if the object is unknown and may have some intrinsic articulations (Liu et al., 2022b). These challenges motivate the development of novel algorithms for the pose estimation and real-time tracking of objects with visuo-tactile data.

Similar challenges also exist for visuo-tactile based object recognition and reconstruction tasks. A notable benefit of shared perception lies in the capability to independently encode feature information for object distinction independently through both visual and tactile sensing modalities. The problem of cross-modal perception is of interest which involves the ability to recognise objects through a secondary sensing modality by leveraging the prior knowledge from a primary sensing modality (Martino and Marks, 2000). For vision-to-tactile cross modal transfer, the state-of-the-art neural networks that have been trained with large-scale dense visual data cannot be directly transferred to sparse tactile data and novel domain adaptation methodologies need to be developed (Falco et al., 2019). Similarly, training a deep neural network requires large annotated training data, which is impractical and time-consuming for tactile data (Liu et al., 2017). Another major challenge is improving the sample efficiency of tactile actions. Some previous work has performed this data collection with manual procedures such as through teleoperation of the robot or following predefined object exploration trajectories (Falco et al., 2019; Vezzani et al., 2016). These approaches are not scalable and require human intervention. Hence, intelligent and autonomous data collection strategies need to be developed in order to improve the efficiency of the process. Previous studies have tackled this challenge by employing information-theoretic strategies to determine the optimal subsequent action based on the current information of the environmental known as active perception (Kaboli and Cheng, 2018). Specifically, within the domain of shared visuo-tactile perception, it is imperative that these tactile information-gathering actions are synchronised with the data acquired from the visual modality. It is evident that efficacy of current methods is limited and the development of novel algorithms are necessary for shared visuo-tactile perception.

1.3 Aims of the Thesis

The primary research question addressed in this thesis is to develop novel methods that enable autonomous robots to integrate visual and tactile sensing to enhance their perceptual understanding of the environment, particularly for tasks such as object pose estimation, recognition and reconstruction tasks. In order to achieve this, a number of objectives have been defined as follows:

I Develop theoretical frameworks capable of estimating the 6 degree-of-freedom (DoF) pose of objects (rigid or articulated) in unstructured scenarios by leveraging the visual and tactile sensing data.

II Develop a methodology for vision-to-tactile cross-modal transfer learning for object recognition enabling robots to leverage tactile sensing under circumstances where vision sensing has been impaired.

III Design methods for reconstructing the shape of unknown objects with visuo-tactile data by leveraging large synthetic datasets for training.

IV Implement the theoretical frameworks on real robotic systems and develop active perception methods to reduce the redundant data collection and improve overall system efficiency.

1.4 Contributions of the Thesis

In addressing these objectives, this thesis presents the following novel contributions:

I A novel method termed the Translation-Invariant Quaternion Filter (TIQF) wherein the pose estimation problem is cast as a recursive filtering problem that handles the sequential and sparse tactile point clouds as well as the dense visual point clouds that are extracted in one-shot manner. By exploiting the geometric constraints in the measured point cloud data, TIQF decoupled the 6 DoF pose estimation problem into rotation and translation estimation and a linear model is developed which is estimated using a quaternion Kalman filter. Additionally, to enhance robustness against local minima and mitigate the dependency on precise initialization, TIQF approach is improved by introducing the Stochastic TIQF (S-TIQF) method, incorporating a robust stochastic initialisation for achieving globally optimal object pose estimation. S-TIQF estimates the 6 DoF pose and 3 DoF scale of unknown instances of categorical objects and relaxes the need for prior known model of the object.

The following publications have resulted from this contribution:

- **P. K. Murali**, M. Gentner, and M. Kaboli, "*Active Visuo-Tactile Point Cloud Registration for Accurate Pose Estimation of Objects in an Unknown Workspace*," IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2021, pp. 2838-2844. https://doi.org/10.1109/IROS51168.2021.9636877.
- **P.K. Murali**, R. Dahiya, and M. Kaboli, "*An Empirical Evaluation of Various Information Gain Criteria for Active Tactile Action Selection for Pose Estimation*," The IEEE International Conference on Flexible and Printable Sensors and Systems (FLEPS 2022), pp. 1–4. https://doi.org/10.1109/FLEPS53764.2022.9781598

II A novel shared visuo-tactile perception method for scene representation and object reconstruction through a data-efficient joint information-theoretic approach for active perception (vision or tactile). The shared and interactive perception is leveraged to find the pose of objects in dense clutter. A scene graph method is introduced for the autonomous decluttering approach that encodes the next object to singulate using either prehensile (grasp) actions or non-prehensile (push) actions. Furthermore, a necessary condition for shared perception is the accurate calibration between the sensing modalities. In this context, a novel approach is proposed for in-situ visuo-tactile based hand-eye calibration using arbitrary objects which removes the constraint of specific hand-eye calibration targets and time-consuming calibration procedures.

The following publications have resulted from this contribution:

- **P. K. Murali**, B. Porr, and M. Kaboli. "*Shared visuo-tactile interactive perception for robust object pose estimation.*" in The International Journal of Robotics Research (IJRR) 2024 https://doi.org/10.1177/02783649241301443.
- **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*," in IEEE Robotics and Automation Letters (RA-L), vol. 7, no. 2, pp. 4686-4693, April 2022. https://doi.org/10.1109/LRA.2022.3150045.

III A novel framework for visuo-tactile-based interactive perception for detecting, tracking and manipulating unknown novel objects (single, multiple, and articulated with revolute or prismatic joints) without assuming any prior knowledge regarding object shape or dynamics. In this regard, a method termed ArtReg (Articulated Registration)

is presented for tracking unknown novel objects (single, multiple, or articulated) by integrating visual and tactile interactive perception with a Manifold Unscented Kalman Filter on the SE(3) Lie Group. ArtReg is deployed for the detection of kinematic chains in objects using a combination of push or hold-pull manipulation actions facilitated by autonomous interactive visuo-tactile perception. Furthermore, the ArtReg pose tracker is also employed within a visuo-tactile based closed-loop control algorithm aimed at achieving precise manipulation of objects towards a specified goal configuration. The full-fledged framework operates effectively under a variety of conditions, including low illumination, visually complex backgrounds, and variations in the centre of mass of the objects.

IV A novel framework for deep active visuo-tactile based cross-modal robotic object recognition is presented in this thesis. The proposed framework consists of three parts: (a) A deep neural network that is trained solely with dense visual point cloud data and tested on sparse point clouds acquired from tactile sensors. A novel unsupervised domain adaptation loss function termed *VTLoss* has been developed for minimising the domain gap between the visual and tactile domain; (b) An active deep learning framework for visual object learning for reducing redundant data collection and annotation; (c) An active tactile-based object recognition approach to reduce the number of tactile actions improving the sample efficiency of the tactile actions.

The following publications have resulted from this contribution:

- **P.K. Murali**, C. Wang, D. Lee, R. Dahiya, and M. Kaboli "*Deep Active Cross-Modal Visuo-Tactile Transfer Learning for Robotic Object Recognition.*" IEEE Robotics and Automation Letters (RA-L), 7(4), 9557-9564. `https://doi.org/10.1109/LRA.2022.3191408`.
- **P.K. Murali**, C. Wang, R. Dahiya, and M. Kaboli, "*Towards Robust 3D Object Recognition with Dense-to-Sparse Deep Domain Adaptation*," The IEEE International Conference on Flexible and Printable Sensors and Systems (FLEPS 2022), pp. 1–4. `https://doi.org/10.1109/FLEPS53764.2022.9781490`

V A novel framework for deep active tactile-based category-level perception of unknown transparent objects for reconstruction termed *ACTOR*. The neural network is trained on a category-level synthetic dataset and tested on sparse tactile point clouds from real unknown transparent objects. The reconstruction of the object model enables downstream tasks such as category-level pose estimation with S-TIQF and object recognition.

The following publication has resulted from this contribution:

- **P. K. Murali**, B. Porr, and M. Kaboli, "*Touch if it's transparent!* ACTOR: Active Tactile-based Category-Level Transparent Object Reconstruction", in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2023. https://doi.org/10.1109/IROS55552.2023.10341680

Furthermore, as part of the doctoral study, a comprehensive review paper on the state-of-the-art of multi-modal sensing and perception methods for in-vehicle applications was conducted, though it is not included in this thesis. It resulted in the following publication:

- **P.K. Murali**, M. Kaboli, and R. Dahiya, "*Intelligent In-Vehicle Interaction Technologies*," in Advanced Intelligent Systems, 2022, 4: 2100122. https://doi.org/10.1002/aisy.202100122.

In addition, contributions have also been made as a co-author to closely related research works in the following publication, which has not been included in this thesis:

- M. Gentner, **P. K. Murali**, and M. Kaboli, "*GMCR: Graph-based Maximum Consensus Estimation for Point Cloud Registration*", in IEEE International Conference on Robotics and Automation (ICRA) 2023. https://doi.org/10.1109/ICRA48891.2023.10161215

This thesis also led to multiple best paper awards at prestigious conference venues as listed below:

- **Outstanding Paper Award** at the 2022 IEEE International Conference on Flexible and Printable Sensors and Systems (FLEPS) (shown in Appendix C, Fig. C.1a).
- **Best Sensors and Perception Paper Award: Finalist** at the 2023 IEEE International Conference on Robotics and Automation (ICRA) (shown in Appendix C, Fig. C.1b).

1.5 Thesis Outline

This thesis is structured into nine chapters as illustrated in Fig. 1.2. **Chap. 2** presents a detailed overview of the state-of-the-art. Drawing inspiration from visuo-tactile perception in humans, this chapter meticulously elucidates the current literature in the domains of visuo-tactile based techniques for pose estimation, object recognition, and reconstruction, as well as techniques pertaining to interactive perception. **Chap. 3** provides details regarding the hardware setup, including the robots and the associated sensors used in the thesis. **Chap. 4** presents the novel TIQF approach for visuo-tactile based object pose estimation.

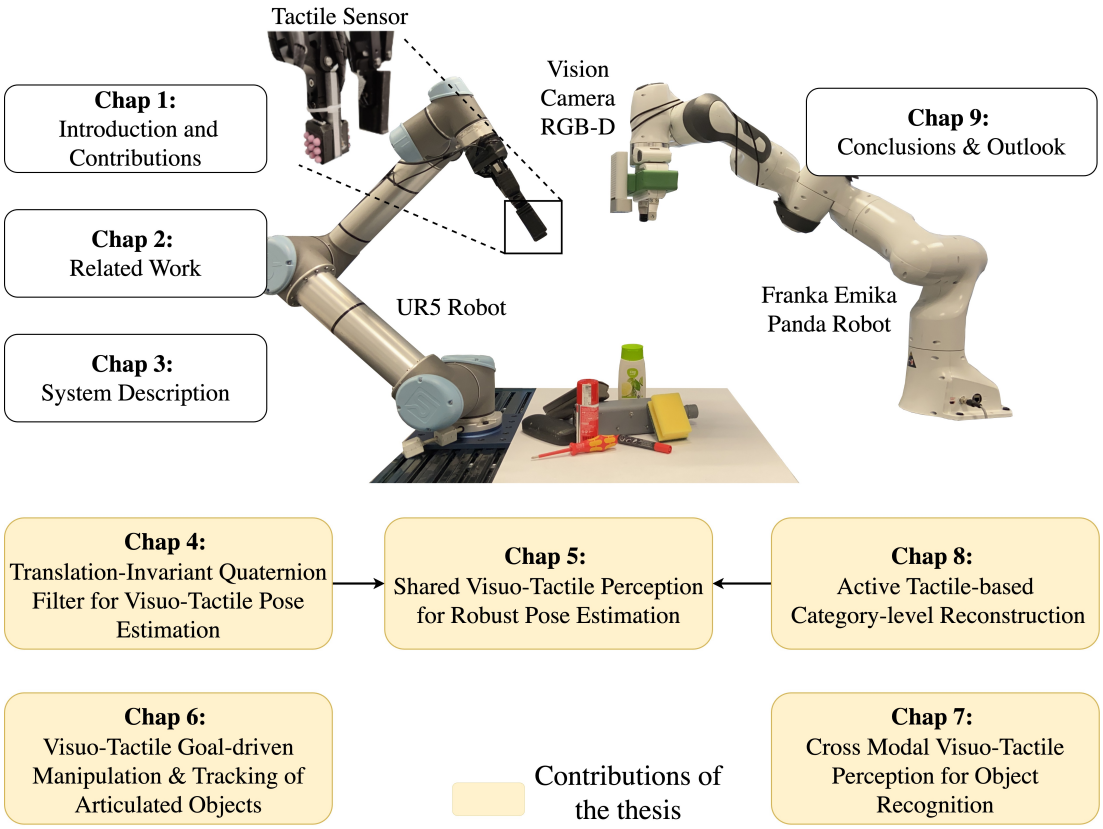


Figure 1.2: The outline of the thesis

Chap. 5 presents the shared visuo-tactile interactive perception approach for robust object pose estimation in dense clutter. The improved S-TIQF algorithm is also presented in this chapter. **Chap. 6** provides details on the novel visuo-tactile based framework for detecting, tracking, and goal-driven manipulation of unknown articulated objects. **Chap. 7** details the novel visuo-tactile based cross-modal perception for object recognition method. **Chap. 8** explains the ACTOR framework for the category-level object reconstruction with tactile data. This methodology presented in this framework is also exploited in Chap. 5 for category-level object pose estimation with visuo-tactile data. **Chap. 9** addresses the aims and provides the conclusions of the thesis and avenues for future research. Since this dissertation tackles distinct tasks involving visuo-tactile perception in robotics such as pose estimation and tracking, object reconstruction, and cross-modal object recognition, each chapter begins with a separate introduction section that details the specific motivation and contributions of the chapter. Each chapter also has a discussion section that specifically summarises the experimental validations and results from the chapter.

Chapter 2

Related Work

This chapter details a comprehensive review of the state-of-the-art methodologies and applications of visuo-tactile sensing within the robotics domain, with a particular focus on its application for pose estimation, object recognition, and manipulation. Initially, this chapter examines the visuo-tactile perception modality in humans, providing a foundation and a compelling rationale for its application and integration within the field of robotics in Sec. 2.1. Although vision-based cameras are ubiquitous, the domain of tactile sensing technology is an emerging and rapidly evolving field, with novel sensor types frequently introduced in scholarly works. Consequently, this chapter presents a detailed review of the prevalent tactile sensing technologies and their respective applications within robotics in Sec. 2.2. Subsequently, the state-of-the-art for visuo-tactile perception is reviewed for pose estimation in Sec. 2.3, object recognition in Sec. 2.4 including reconstruction techniques and finally visuo-tactile interactive perception wherein prehensile and non-prehensile methods are presented in Sec. 2.5. In each section (Sec. 2.3-2.5), the open challenges are systematically identified in the literature, elucidating how this thesis seeks to address certain unresolved questions.

2.1 Visuo-Tactile Modality in Humans

The human brain seamlessly integrates data from various sensory modalities to evaluate and comprehend the environment and its constituent objects (Hillis et al., 2002). Human visual sensing plays a major role for scene understanding (Pei et al., 2021). Tactile or haptic perception is often used to supplement visual perceptual information (Hillis et al., 2002). Both perceptual systems provide complementary information, as seen in Fig. 2.1. The visual perception provides global scene information such as colour, brightness information, shapes, and pose (position/ orientation) of objects around us. However, tactile

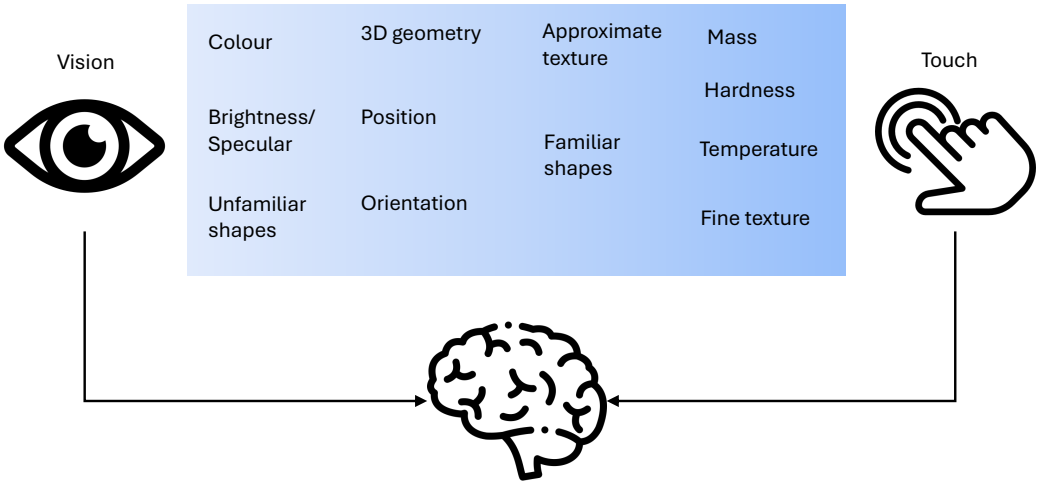


Figure 2.1: Information pertaining to an object may either be specific to a particular modality or shared across multiple modalities. In this figure task-dependent modality biases are dilineated. For instance, only visual perception can detect color information (left), whereas tactile perception is uniquely capable of detecting mass (right). Conversely, certain attributes, such as orientation, are accessible to both modalities. Nonetheless, each modality may differentially modulate the perception of these shared features. This figure is adapted from (Woods and Newell, 2004) with modifications.

sensing provides fine-grained local information such as local texture, hardness, mass, temperature, and so on. Furthermore, the tactile modality is sequential in nature for observing object properties, whereas the visual modality is instantaneous with the ability to perceive the surroundings in a single glance. The human skin being the largest sensory organ of the body, it increases the limited field of view of the eyes to improve our perceptual understanding (Toprak et al., 2018).

The development of tactile sensory capabilities commences remarkably early in neonatal development, with infants demonstrating the acquisition of tactual and motor skills, such as the behavior of mouthing objects (Rochat et al., 1988). It was found that infants of 2 to 5 months of age use tactile perception alone to discriminate object shapes (Streri and Pêcheux, 1986; Streri, 1987). Streri (1987) found that babies become acquainted with the shape of objects tactually and explore them for decreasing periods of time, and this without any visual information. This habituation is similar to that obtained in the visual situation. Striano and Bushnell (2005) empirically found that 3-month-old infants were able to discern objects of varying shapes, texture, compliance and weight using touch alone. Lederman and Klatzky (1987) also defined a set of exploratory procedures (EPs) for human grasp behavior. A detailed survey of the extant literature on neonatal tactile perception is accessible in this comprehensive review (Streri and Milhet, 1988).

Similar to tactile perception studies, visual perception and visuo-motor coordination studies have been done for human infants and animals alike. Held and Hein (1963) analysed the development of visually guided behaviour in kittens. They found that this development critically depends on the opportunity to learn the relationship between self-produced movement and concurrent visual feedback. The authors conducted an experiment with kittens that were only exposed to daylight when placed in a carousel. Through this mechanism, the active kittens transferred their own deliberate motion to the passive kittens that were sitting in a basket. Although both types of kittens received the same visual stimuli, only the active kittens showed meaningful visually guided behaviour in test situations. Later, Hein and Held (1967) extended their kitten study and found that kittens deprived of the visual feedback of their front limbs had an impaired reflex on uneven surfaces. In humans, we know that visual feedback is critical for interaction and manipulation. In human adults, visuo-motor skills are highly refined and integrated with tactile feedback for fine-manipulation. Researchers have studied human infants in the early stages of development in order to investigate the role of visual feedback for motor actions. Before 4-5 months of age, the arm movement of an infant is ballistic i.e., it may be triggered by vision but it lacks corrective actions that may be provided with visual feedback (Bower, 1974). This notion was substantiated in Bower's study, where 5-month-old infants were given objects within their reach, and subsequently, the lights in the room were turned off to create complete darkness (Bower, 1972). The approach attempts were not affected by the lack of visual feedback. Infants between 5 and 9 months perform vision-guided movements. This was found in the studies in McDonnell (1975), wherein babies aged 4 to 10 months wore prismatic glasses that shifted the object by 7 cm. Younger babies were unaffected and did not alter their grasping paths whereas older babies altered their hands once they missed the target. The studies suggest that for younger babies ($<$ 5 months) the visuo-motor control is performed in open-loop like fashion whereas for older babies motor control is visually guided. In adults, there is a combination of ballistic and fine grained motions which is attributed to better motor control by (Bushnell, 1985). Jeannerod (1984) observed that in adults, there is a distinct reaching phase and correction phase prior to prehensile manipulation actions. The reaching phase is a directional movement to the visual target, whereas the fine adjustment prior to prehension (grasping) is a response to object properties. The studies showed that the subjects were not affected for the two phases with and without visual feedback. Clearly, as humans grow older the motor skills become more mature and the visual perception is integrated with tactile perception.

The process by which the human brain integrates visuo-tactile information has garnered substantial research attention. Ernst and Banks (2002) postulated that humans combine

visual and tactile perceptual information in a statistically optimal manner following the maximum likelihood estimation. The neurons in the cortical region of the inferior parietal cortex integrate both the large-scale visual space and the fine-scale tactile space represented by the hand, thereby preventing errors due to environmental complexities such as changes in pose of objects (Wan et al., 2020). Studies have shown that this ability of visuo-tactile integration is not inherent in the brain and is developed by experience accumulated during development (Miikkulainen et al., 2006). Furthermore, studies have shown that this ability to integrate visuo-tactile perceptual information in a statiscally optimal manner is absent in children below 8 years of age and only starts to develop between 8-10 years of age (Gori et al., 2008). More critically, it is known that the integration of the two modalities becomes more likely if it is known that the stimuli come from the same object (Helbig and Ernst, 2007). Intermodal transfer between vision and touch is of particular importance for effective fine manipulation of objects. It is important to note that vision modality is for spatial perception only whereas the touch modality can be used both for perception through exploration and as a *functional modality* capable of interacting and modifying the environment during perception. Bernstein (1967) argues that the motor system is the integrative link between vision and touch. In fact, for a functional action to succeed, the motor actions that move the hand should correspond to a visually seen target. Furthermore, Van der Kamp et al. (2003) draws the distinction between perception-oriented actions and goal-oriented actions. For instance, perception-oriented actions such as running fingers on the edges of objects, rotating objects and other exploratory actions whereas in goal-oriented actions such as locating the pose of objects, the perception is only used to facilitate the goal. Once the object pose is extracted, exploration stops and any additional information gained is ignored. Körding and Wolpert (2004) postulated that the central nervous system follows Bayesian integration for sensorimotor learning. Bach-y Rita and Kercel (2003) proposed that the act of perception is performed in the brain and not in the sensory organs. It is the brain's *plasticity* or the ability to adapt to changes in the external stimuli which allows for inter-modal transfer and sensory substitution.

2.2 Visuo-Tactile Perception: From Humans to Robots

From birth, human infants learn to understand and explore the world by seamlessly integrating the vision and touch sensing together with motor control for the limbs (Held and Bauer, 1967). From a conceptual level, the human perception system has been the inspiration for machine perception. Considering robot perception, the vision sensing technologies such as RGB/ RGB-D cameras, time-of-flight (ToF) laser sensors, structured-light laser sensors, infrared sensors, event-based cameras and so on have become ubiquitous and

easily commercially available (Horaud et al., 2016). In comparison, tactile sensing technology does not have a one-solution-fits-all and each type of sensor is designed for specific applications (Dahiya et al., 2009). Adult humans have about 45000 mechanoreceptors responsible for touch sensing embedded in the skin with different spatial density on various parts of the body (Taube Navaraj et al., 2017). For instance, the fingertips have about 241 mechanoreceptors per unit centimetre whereas the palm has about 58 only (Johansson and Vallbo, 1979). The spatial acuity, which is the smallest distance to where a person can distinguish two points, is about 1mm on the fingertips and 30mm on the belly (C. Craig, 1998). In addition to pressure and vibrations sensed by mechanoreceptors, the human skin can also sense temperature through thermoreceptors and pain through nocioceptors (Johansson and Westling, 1984). The skin can also detect stimuli with more than 500Hz frequency (Dahiya et al., 2009). Hence, the human tactile skin has been an inspiration for researchers developing tactile sensors for robotics. Artificial tactile sensors mainly mimic the mechanoreceptors of the human skin with the objective to measure the external force at multiple contact points (Navarro-Guerrero et al., 2023). Thermoreceptors are typically not included with tactile sensors for robotic applications. However, there are some exceptions where temperature sensors were integrated with pressure sensors to compensate for the thermal effects (Wade et al., 2017). Similarly, nocioreceptors that are integral part of the human tactile system are typically not used for robotic applications. Commercial tactile sensors can measure 3D force at each taxel, with good spatial resolution (about 1.6 taxels/cm^2) and high sampling rates ($>$ 100Hz) (Navarro-Guerrero et al., 2023). Moreover, large-area tactile sensors that cover the embodiment of robots are becoming increasingly popular in research (Dahiya et al., 2019). For instance, in some multi-finger robotic hands, the finger tips have higher spatial resolution of sensors compared to the palm region (Taube Navaraj et al., 2017). In the following section, a brief overview of the types of vision and tactile sensors that are commonly used in robotics is presented.

2.2.1 Vision Sensing Technology

Cameras that provide vision-based sensing are the primary sensing modality for robots for perception, locomotion, and manipulation tasks. Typically, monocular cameras providing RGB images are used in conjunction with image processing algorithms. More often, 3D depth information is required for variety of applications, and the common depth sensors are structured light, time-of-flight (ToF) cameras and stereo-cameras (Horaud et al., 2016). Structured light depth cameras project a coded pattern onto the surface of the object which that is modulated by the distance of the object from the camera which is used to recover the depth image (Zanuttigh et al., 2016). ToF cameras actively emits light and measures the

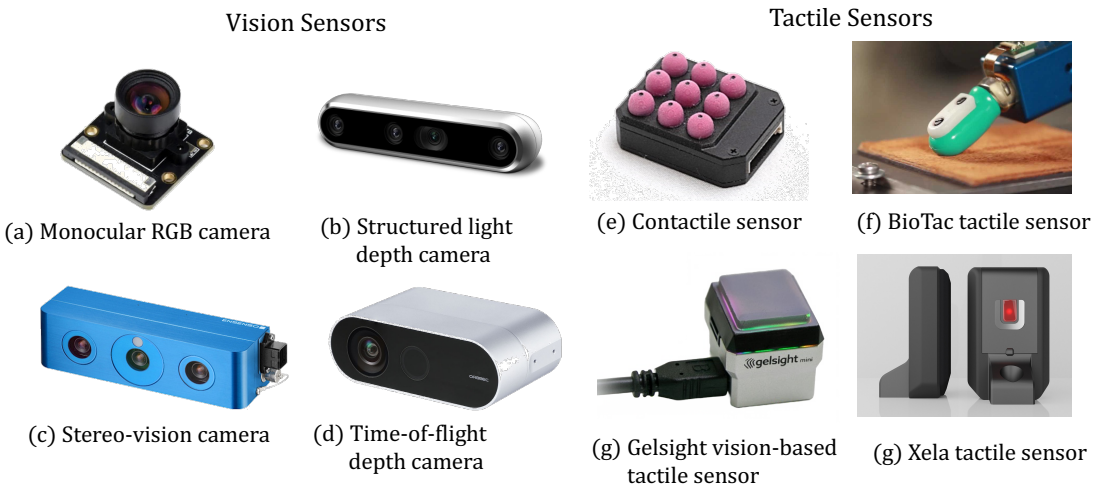


Figure 2.2: Some examples of vision and tactile sensors: (a) monocular RGB camera, (b) Intel Realsense structured-light depth camera and RGB camera, ©Intel, (c) Ensenso stereo-camera, ©IDS Imaging, (d) Orbbec time-of-flight camera, ©Orbbec Inc., (e) Contactile sensor, ©Contactile, (f) Biotac tactile sensor, ©SynTouch, (g) Gelsight vision-based tactile sensor, ©Gelsight, (g) Xela tactile sensor (Tomo, 2019), ©Xela Robotics. Reproduced with permissions.

time delay or phase delay of the reflected light to recover the depth information (Horaud et al., 2016). Stereovision cameras use two cameras to capture images of the same object. The depth information is recovered by calculating the disparity map with the triangulation method (Grosso et al., 1989). The Tab. 2.1 provides a summary and comparison of the commonly used vision sensing technologies. The Fig. 2.2 shows some of the commonly used vision sensors in robotics.

2.2.2 Tactile Sensing Technology

Tactile perception and the associated information processing techniques are tightly coupled with the tactile sensing technology that is used. There is a distinction between tactile perception and tactile sensing. Tactile perception refers to the abstract cognitive-level organisation of rich sensor information whereas tactile sensing is the measurement of physical properties and the encoding of raw physical signals into low-level digital inputs (Li et al., 2020a). Several review papers are available that discuss the development of tactile sensing technology (Nicholls and Lee, 1989; Lee and Nicholls, 1999; Dahiya et al., 2009; Cutkosky and Provancher, 2016; Kappassov et al., 2015). Typically, the principle of tactile sensing technology varies based on the type of transduction technology employed. The change in capacitance, resistance, optical distribution, electrical charge are usually used to detect contact (Kappassov et al., 2015). Some of the commercially available tactile sensors

Table 2.1: Summary and comparison of common vision sensing technologies.

Sensor Type	Sensing Principle	Advantages	Disadvantages
Monocular Camera	* Uses convex lens to focus scenes on the camera's image plane * 2D image output	+ High resolution possible + High framerates possible (15- 600fps) + Easy software interfacing + Features easily extraction	- Sensitive to ambient light - Sensitive to occlusions
Structured Light Depth Camera	* Projects coded optical patterns to objects and measures patterns' variation * 3D depth output	+ Typical lower framerates (30-60fps) + Higher precision + Cost effective and compact size + High stability	- Sensitive to object transparency - Sensitive to reflections - Sensitive to ambient light - Higher processing time - Limited working range
Time-of-Flight (ToF) Depth Camera	* Directly measures the time delay or phase delay of the reflected light * 3D depth output	+ Faster real-time performance + Wide working range + Robust to ambient light + Higher frame-rates possible	- Lower precision - Multipath interference - Higher cost & power consumption - Sensitive to transparency, reflections
Stereo-Vision Camera	* RGB image feature point matching and indirect triangulation calculation * 3D depth and Image output	+ No need for external lightsource + Cost effective + Simple hardware requirement	- Compute intensive - Limited depth range - Reliance on texture - Sensitive to light, transparency

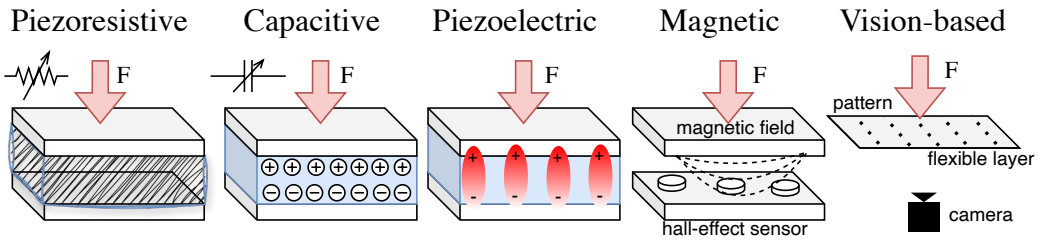


Figure 2.3: Schematic illustration of working principles for the popular tactile sensor types used in robotics.

are shown in Fig. 2.2. Schematic illustrations of commonly used tactile sensing technology are shown in Fig.2.3.

Piezoresistive sensors exhibits a variation in resistance when subjected to an external force. These have been used in force sensing resistors (FSR), pressure-sensitive conductive rubber, piezoresistive foam, and piezoresistive fabric (Drimus et al., 2014; Büscher et al., 2015). These sensors are easy to manufacture and can be made flexible. However, they suffer from non-linear response and low repeatability after multiple deformations. Many commercial solutions exist such as Weiss tactile sensors (Weiss, 2025) and ATi industrial automation (ATI, 2025).

Capacitive sensors measure force based on change in capacitance caused by variation in the gap between the parallel plates of a capacitor on application of force. Normal and shear forces can also be measured by capacitive sensors (Lee et al., 2008). Highly sensitive sensors can be designed by using more compressible materials or thin sensors. The capacitive sensors are very popular for use in mobile devices as touch screens as well as for use in robotic applications (Dahiya et al., 2009) such as in iCub humanoid robot skin (Schmitz et al., 2011) and the Allegro hand (Jara et al., 2014). The major drawback are the susceptibility to electro-magnetic noise, sensitivity to temperature and non-linear response (Dahiya and Valle, 2012).

Inductive or magnetic-based sensors wherein magnetic field changes are converted to forces are capable of providing 3-axis force measurements that can be devised into array-like structures. For instance, Tomo et al. (2017) developed the XELA uSkin sensor, a 3-axis force sensing soft dome fingertip made from flexible PCB, covered by a silicone skin embedded with magnets. However, these sensors cannot be used near eletromagnetic equipment that can alter the magnetic field.

Piezoelectric sensors are sensors where the electric charge that is produced on application of force is used for dynamic tactile sensing. For instance, Polyvinylidene fluoride or polyvinylidene difluoride (PVDF) material has been used for touch sensing (Seminara

et al., 2011). These sensors have high bandwidth upto 7kHz as reported by (Goger et al., 2009).

Barometric pressure tactile sensors measures the external force by the pressure of a fluid that also gives deformability and softness ability to the sensor. For instance, the BioTac sensor from SynTouch LLC consists of a fluid pressure measurement electrodes that detect micro-vibrations (Fishel and Loeb, 2012). The BioTac sensor consists of *multi-modal sensors* that can measure force, vibrations and temperature. Similar to human skin that can detect multiple modalities, there has been considerable work in order to develop multi-modal electronic skin (e-skin) (Dahiya et al., 2019). For instance, Jung et al. (2020) presented a flexible tri-mode sensor stimulated by human skin with three different vertically stacked sensors, i.e., a hair-type flow sensor, temperature sensor, and pressure sensor.

Acoustic ultrasonic-based sensors have also been reported to be used as tactile sensors for detecting onset of motion and slip in grasping experiments. Jiang and Smith (2012) presented a novel acoustic-based tactile sensor for pre-touch called the seashell effect pre-touch sensor. The seashell-effect sensor is an open ended pipe with a microphone and detects changes in the spectrum of ambient noise that occur when the sensor is approaching an object. In (Ando and Shinoda, 1995), the authors designed a 2x2 array of PVDF that is used to sense and localise contact point from ultrasonic pulse emissions.

Quantum tunnel effect sensors have also been used for tactile sensing applications wherein quantum tunnel composites (QTC) that on application of external pressure can turn from an insulator to a conductor are used. Using QTC material, Zhang et al. (2012) developed a flexible tactile sensor for an anthropomorphic artificial hand with capability of measuring shear and normal forces. QTC was integrated in the previous version of the Shadow robotic hand as well (Walkler, 2004). They exhibit linear response however they suffer from wear and tear and the sensitivity reduces over usage. Wiring is a typical problem for large area tactile sensing, thus limiting the design of tactile sensing skins for specific embodiments.

Electrical Impedance Tomography (EIT) is one method which has gained attention for designing tactile sensors (Kato et al., 2007; Alirezaei et al., 2009; Yao and Soleimani, 2012; Silvera Tawil et al., 2012). EIT is an imaging technique that uses electrodes to measure the voltage or current signals from the boundaries of the electrical conductors and reconstructs the internal conductive distribution. It can be used to implement robotic tactile sensing if a flexible stretchable material is used to develop the electrodes and attached to the robot surface (Liu et al., 2020). Silvera-Tawil et al. (2014) provided a comprehensive reivew of EIT techniques used for designing tactile sensors. Key features of EIT based tactile sensors that make it a viable option for designing large area sensors:

Table 2.2: Summary and comparison of various tactile sensing technologies.

Sensor Type	Sensing Principle	Advantages	Disadvantages	Example References
Piezo-resistive	Change in resistance	+ Can be flexible + Many form factors: fabric, array, foam, conductive rubber + Can be used for large area + Commercial sensors available	- Hystersis - Permanent deformation - Non-linear response - Temperature and moisture dependence	Drimus et al. (2014) Büscher et al. (2015)
Capacitive	Change in capacitance	+ Can be flexible + Dense array design possible + Commercial sensors available	- Static measurements only - Stray capacitive measurements - Hysteresis	Schmitz et al. (2011) Lee et al. (2008)
Piezo-electric	Change in electric charge	+ Dynamic tactile sensing + High dynamic range, high sensitiveity	- Temperature sensitivity - No static sensing - Poor spatial resolution	Seminara et al. (2011) Goger et al. (2009)
Vision-based tactile sensing	Light intensity/ spectrum change	+ High sensing range, high spatial resolution + High accuracy, high repeatability + Rapid response + Immune to EMI	- Bulky, non-conformable - High power requirement - High computation cost	Johnson et al. (2011) Yamaguchi and Atkeson (2017)
Triboelectric-based	Change in mechanical energy to electrical energy	+ Can be highly flexible and stretchable + Self-powered sensing device + Comparable sensitivity and dynamic range + Easy fabrication, low cost	- Reliable calibration necessary for quantitative measurements - Material susceptible to humidity	Cheng et al. (2019)
Inductive/ Magnetic	Change in Magnetic coupling	+ Linear model/ output + High dynamic range	- Moving parts, bulky - Susceptible to EMI	Tomo et al. (2017)
Electrical Impedance Tomography	Change in Electrical Impedance	+ No internal wiring of sensing part, stretchable, flexible. + Continous sensing capability + High repeatability, range of sensing. high scalability	- Low spatial resolution - Low temporal frequency - Requirement of continuous energy input	Kato et al. (2007) Alirezaei et al. (2009)
Electrical Capacitance Tomography	Change in capacitance	+ Provides quantitative measurements compared to EIT + Provides proximity and contact information	- Not widely used in robotic applications - Requirement of continuous energy input	Mühlbacher-Karrer et al. (2017)
Barometric pressure measurement	Change in fluid pressure	+ High bandwidth, high sensitivity + Temperature and moisture independence	- Low spatial resolution	Fishel and Loeb (2012)
Photovoltaic-based sensing	Change in light intensity	+ Self-powered sensing device + No explicit sensing layer, solar cells used as intrinsic sensors + Shape and proximity sensing	- No pressure sensing - Poor spatial resolution - Sensitive to ambient light	Escobedo et al. (2020)
Quantum tunnel effect	Quantum tunnelling	+ Linear response + Relatively high dynamic range	- Complex manufacturing process	Zhang et al. (2012) Walkler (2004)
Ultrasonic-based	Ultrasonic pulse-echo ranging	+ Can be used to detect slip and onset of motion + Fast dynamic response + Good force resolution	- Difficult to integrate in miniaturized circuitry.	Jiang and Smith (2012)
Multi-modal	Multiple sensing components	+ Static and dynamic pressure sensing, temperature measurement and proximity sensing + Can achieve high spatial resolution	- Bulky size due to various components	Fishel and Loeb (2012) Jung et al. (2020)

- sensing area is composed of homogeneous material without any or limited wiring necessary
- flexible and stretchable material can be used
- since only one single material is used, it can provide continuous sensing
- since the response of the system depends on the localised conductivity changes of the variable-conductance material in response to an external stimulus, materials sensitive to different types of stimuli, such as temperature, could be used to sense other types of excitation

Similar to EIT, Electrical capacitance tomography (ECT) is another imaging method for visualizing permittivity distribution in the interior of a dielectric object by measuring the capacitance at the boundary. ECT can provide a quantitative measurement whereas EIT can provide a qualitative measurement only (Liu et al., 2020). Although not widely used in robotics, few works such as Mühlbacher-Karrer et al. (2017) demonstrated the use of ECT for robotic object detection. On the issue of power requirements, for autonomous robots to perform in unstructured environments, variety of sensors, actuators and compute is necessary that significantly raises the energy requirements. Triboelectric energy generators (TENG) which generate energy by touching, pressing, twisting, stretching, etc., are particularly interesting for e-skin design (Dahiya et al., 2019). TENG has the advantages of low-cost, easy assembly and high output voltage. TENGs composed of stretchable polymers exhibit high stretchability and bendability and can be woven into textiles as well. Cheng et al. (2019) developed a highly stretchable TENG-based sensor and demonstrated the sensing ability for dynamic force, temperature, and location detection, and furthermore, the self-powered capability of the sensor.

In contrast to the aforementioned sensing technologies, *Vision-based tactile sensors* such as the GelSight sensor (Johnson et al., 2011) are also popular which are composed of an elastomer coated with a reflective membrane and a camera with resolution of up to 2 microns. Due to the high resolution in camera input, the resulting force field resolution is higher than any traditional force sensors (Johnson et al., 2011). The sensor measures normal force, in-plane torque, tilt torque, shear and slip (Yuan et al., 2015). Similarly, Lepora and Ward-Cherrier (2015) designed the TacTip family of sensors that are 3D printed fingertips with a camera inside. Pins in a regular pattern line in the interior of the fingertip, which are observed by the camera for pin positions and deflections. The fingertip is filled with silicone gel to give it compliance. Another vision-based tactile sensor is the FingerVision by Yamaguchi and Atkeson (2017) which is a transparent sensor that seeks to

measure proximity, vibration, and force, using image processing. Kumagai and Shimonomura (2019) designed an event-based tactile sensor consisting of a elastomer with markers and an event-based camera that detects temporal changes. Similarly, Alspach et al. (2019) designed Soft-bubble a vision-based tactile sensor which uses a depth sensor instead of a traditional camera to detect contact, force and slip estimation. In the domain of optical sensors, photodiodes have been used to capture the deflection of illuminated infrared light upon application of external force as in the Contactile sensor (Khamis et al., 2018). A summary of various tactile sensing technologies and the advantages and disadvantages is detailed in Tab. 2.2.

2.3 Visuo-Tactile based Object Pose Estimation

In order to manipulate objects and understand the world, a robot needs to recognise and identify the 6 Degree-of-Freedom (DoF) pose of the objects. Accurate and timely estimation of pose is critical, as even small inaccuracies can lead to grasp failures (Luo et al., 2017). Typically, the pose of an object is represented by the position and orientation of the object in the coordinate frame of the world or the robot base. Object pose estimation is a well-researched domain within computer vision, predominantly utilizing vision sensors that capture the entirety of the scene in a single acquisition. However, vision-based pose estimation suffers from inaccuracies due to incorrect calibration of the sensors, environmental conditions (occlusions, presence of extreme light, and low visibility conditions), and object properties (transparent, specular, reflective). Multi-modal object pose estimation wherein one modality can compensate and verify the estimation results have received attention (Lee et al., 2020). As in humans, high fidelity tactile sensing can be used to augment, compensate and verify vision-based estimation in autonomous robots. In this section, the current state-of-the-art methods in tactile-based and visuo-tactile based object pose estimation techniques are detailed.

Point cloud registration is a common technique for 6 DoF pose estimation and its the process of finding the rigid transformation that aligns two point clouds. Typically in the pose estimation context, it involves the registration of the point cloud extracted from the CAD model of the object of interest with the sensor-acquired point cloud. Furthermore, information between vision and tactile data can be fused through point clouds of the objects as raw point clouds are sensor agnostic (Rusu and Cousins, 2011). When the correspondences between the two point clouds are known *a priori*, the registration problem can be solved in closed form (Horn, 1987). However, correspondences are unknown in practical situations, and standard approaches involve iteratively finding the best correspondence and the transformation given the best correspondences known as the iterative closest point (ICP)

algorithm (Besl and McKay, 1992). ICP and its variants (Pomerleau et al., 2013) are batch registration methods that are known to have low performance when provided with sparse point cloud data that arrive sequentially as is the case with tactile measurements (Glozman et al., 2001). In contrast, other approaches relied on finding dense point-to-point corresponding using feature extraction and then optimised for 6D pose (Yang et al., 2020; Rusu et al., 2009; Huang et al., 2021b). Recently, deep learning approaches have been used to learn robust features for generating correspondences followed by an optimization such as RANSAC (Zeng et al., 2017; Deng et al., 2018). Deep learning approaches have also been used to regress the pose directly using an end-to-end approach. This is done by learning to regress the pose parameters directly from the features of the input point clouds (Yang et al., 2019; Pais et al., 2020; Huang et al., 2021a).

Filter-based approaches were also generally preferred for sequential data as in the case of tactile data. Petrovskaya and Khatib (2011) developed a novel particle filter, named Scaling Series, which localises the object using touch sensing efficiently (under 1s) and reliably ($\geq 99\%$). In (Vezzani et al., 2017), the authors proposed a novel filtering algorithm called the Memory Unscented Particle Filter (MUPF) drawing inspiration from the Unscented Particle Filter (UPF) (Van Der Merwe et al., 2001) to localise the object recursively given contact point measurements. Alongside contact point measurements based localization, array-based tactile sensors have been used to extract local geometric features of objects using principal component analysis (PCA) in order to localise the object by matching the covariances of the extracted tactile data and object geometry (Bimbo et al., 2016). Vision has been used to provide an initial estimate of the object pose that was refined by tactile localisation using local or global optimization techniques (Bimbo et al., 2015; Hebert et al., 2011).

Analogously to SLAM approaches in mobile robots, in (Yu et al., 2015) SLAM has been applied to recover the shape and pose of a movable object from a series of observed contact locations and normal contact with a pusher. Similarly, Suresh et al. (2020) approached the problem of simultaneous estimation of shape and pose of a planar object. They used Gaussian process implicit surface (GPIS) regression for shape inference and factor graphs for pose estimation. However, compared to exploration using visual feedback, tactile exploration is challenging in the sense that touch sensing is intrusive in nature, that is, the object (environment) is moved by the action of sensing (Luo et al., 2017). Depending on the shape complexity of the object, higher resolution of tactile measurements allows to extract distinctive features of the object in order to aid its recognition and pose estimation. Vision-based tactile sensors such as GelSight sensor have been used to localise in-hand objects by collecting a series of images and creating a heat-map as in (Li et al.,

2014). Keypoint extraction was performed on the tactile images and matched using image registration methods. Recently, Kuppuswamy et al. (2019) used soft-bubble sensors which has an elastic membrane and an internal ToF based depth sensor (Alspach et al., 2019) in order to estimate the pose of objects. They modelled the deformation of the membrane when contacted with the object and devised an ICP-based pose estimation pipeline based on the contact patch measurements. Hebert et al. (2011) presented a technique for in-hand localisation of objects by fusing the vision, tactile and force/torque sensor information. Bhattacharjee et al. (2015) postulated that visually similar surfaces should also have similar haptic properties. Based on this fact, they created a dense haptic map efficiently across visible surfaces with sparse haptic labels which was used for grasping experiments in cluttered environments.

Contrary to instance-based methods, recent works have addressed the pose estimation of unknown objects from known *object categories* without any prior instance-specific 3D CAD models available which is known as category-level object pose estimation. Wang et al. (2019) introduced the problem of category-level pose estimation and presented the Normalised Object Coordinate Space (NOCS) that produces a shared canonical representation for all object instances in each category. The predicted NOCS map is used to extract the pose and shape of objects with the observed depth map. Lee et al. (2021) extended the NOCS map with a CNN-based category level pose estimation with RGB images with little or none depth information. Similarly, other works have used variational auto-encoders (VAE) for generating the canonical 3D point clouds and the pose is regressed using another deep neural network (Chen et al., 2020). Some works explicitly model the intra-class shape variations using deformation from pre-learned shape priors (Tian et al., 2020). In addition to pose estimation, Deng et al. (2022) combined their category-level auto-encoder with a particle filter framework for tracking of unknown objects in an iterative manner. The method relies upon accurate depth estimation and semantic segmentation as input. Similarly, Wen and Bekris (2021) performed 6D pose tracking for unknown objects using learnt networks for segmentation and keypoint extraction and pose graph optimisation for pose tracking. As the accuracy of category-level estimation is far from satisfactory in comparison to instance-level methods, some works perform iterative point cloud pose refinement after finding the categorical shape prior (Liu et al., 2022c). While manipulating objects in-hand, the objects are typically occluded from the line-of-sight of the camera. Prior works have fused vision and tactile sensing data to accurately measure and track the pose of in-hand objects using Bayesian filtering techniques and deep learning methods (Álvarez et al., 2019; Dikhale et al., 2022; Pfanne et al., 2018). Recent works used deep learning based approaches along with pose-graph optimization to track and recover the shape of novel ob-

jects during in-hand manipulation by combining visual and tactile sensing (Qi et al., 2023).

While the aforementioned studies focused on rigid objects, complex objects may also have articulations such as drawers, reading glasses, microwave ovens, and so on. Although there are computer vision techniques for tracking articulated objects (Lowe, 1991; Nickels and Hutchinson, 2001; Pellegrini et al., 2008; Schmidt et al., 2015; Liu et al., 2022b; Kanazawa et al., 2018; Ge et al., 2019), detecting and tracking unknown articulated objects without damaging them required the integration of tactile sensing. Some works such as (Sturm et al., 2011; Martín-Martín and Brock, 2022) have developed interactive perception system with visual and force/torque sensing but have relied upon known model, marker-based systems for pose tracking and hand-crafted features that limit the generalisability of the system.

While tactile data can be collected in a randomised manner or driven by a human-teleoperator (Vezzani et al., 2017), in order to reduce the time and the amount of measurements, active strategies are required that involve the generation of candidate actions and the selection of the best next action according to expected *information gain*. Hebert et al. (2013) used an information gain metric based on the uncertainty of the object's pose to determine the next best touching action to localise the object. Hsiao et al. (2010) implemented a decision-theoretic approach and an approximate Partially Observable Markov Decision Process (POMDP) to select actions for exploration. Saund et al. (2017) also use an information gain approach for localisation using a particle filter. Similarly, Tosi et al. (2014) approached the "next best touch" problem for object localisation by also considering an information gain approach and additionally, constraining the optimization problem by including the computation and action execution time. However such constraints are application specific and limits the generalisability of the approach.

2.3.1 Limitations in the state-of-the-art

Current pose estimation/ 3D registration methods in the literature perform poorly with very sparse point clouds ($10-100$ points) (as pointed out by several works such as (Agamennoni et al., 2016; Tazir et al., 2018; Razlaw et al., 2015)) and new algorithms need to be developed that can handle the point sparsity as well as sequential nature of the input data. Additionally, such methods need to also perform accurately for dense point clouds from visual sensors that capture the entire scene in one-shot (non-sequential). When robotic systems are equipped with multi-modal sensing such as vision and tactile sensing, sharing the perceptual information gathered from the same scene through different sensing modalities is crucial for the robustness of the system. The existing literature often employs visuo-tactile sensing sequentially, for example, to verify the estimation with one modality of

the other modality (Bimbo et al., 2015; Hebert et al., 2011) and lacks methodologies that facilitate the sharing of visuo-tactile perception, a critical aspect addressed in this thesis. Furthermore, it is well known that vision-based pose estimation is sensitive to occlusions and localising objects in clutter (Luo et al., 2017). Tactile perception inherently constitutes an active modality, which can be strategically leveraged to perform interactive perception, thereby enhancing the efficacy of visual perception (Li et al., 2020a). Moreover, objects can be also have inherent articulations and may move while performing probing actions by the robot. Current state-of-the-art focuses on pose tracking of articulated objects with computer vision techniques as in (Ge et al., 2019; Kanazawa et al., 2018; Schmidt et al., 2015) and techniques that can leverage vision and tactile sensing to detect and track the pose of articulated objects in real time are lacking in the current literature. Finally, improving the sample efficiency of tactile actions is crucial for an efficient robotic system and active perception techniques that leverage the current known information gathered from either visual or tactile sensing to decide on next tactile actions are needed (Tosi et al., 2014). These issues are addressed in this thesis in Chaps. 4-6.

2.4 Visuo-Tactile based Object Recognition & Reconstruction

The task of object recognition represents another fundamental challenge within the domain of robotics. With the advent of deep learning paradigms and large annotated image datasets, the computer vision domain has experienced substantial advances in object recognition methodologies (Han et al., 2019; Andreopoulos and Tsotsos, 2013). However, this issue remains an unresolved challenge for embodied robotic systems when faced with novel unseen objects in real-world environments (Yang et al., 2018). Oftentimes, shape reconstruction of unseen objects is a necessary task prior to performing recognition of pose estimation. Objects with shiny or transparent surfaces pose a challenge for vision sensors for the task of recognition or reconstruction (Wang et al., 2022a). Tactile sensing provides complementary information to visual data and is not affected by specularity or transparency (Li et al., 2020a). Hence, many works in literature have leveraged the tactile modality in a monomodal fashion and multi-modal together with visual sensing for object recognition and reconstruction tasks (Kaboli et al., 2019, 2017; Liu et al., 2016a; Abderrahmane et al., 2018). This section details some of the relevant works in the state-of-the-art with particular focus towards multi-modal visuo-tactile based approaches.

Tactile sensing alone has been used for the task of object recognition in literature. Novel descriptors that extract the statistical properties of tactile signals in the time domain have

been used to identify and discriminate objects and various textures (Kaboli et al., 2017; Kaboli and Cheng, 2018; Kaboli et al., 2019). In other works, the sparse point clouds from tactile sensing were used to approximate the shape of the object using superquadratics and then Gaussian processes were used to classify the objects (Jin et al., 2013). Similarly, covariance of contact image from a tactile sensor were used as a feature descriptor with a Naive Bayes classifier for object recognition (Liu et al., 2012). Some works also leveraged classical computer vision feature descriptors such as SIFT to extract feature from the tactile heatmaps (Pezzementi et al., 2011). Earlier works also developed a bag-of-features based approaches for object classification based on low resolution tactile images. Kernel sparse coding based techniques have also been used for tactile based classification tasks (Liu et al., 2016a). Liu et al. (2017) provides an extensive review on recent developments in tactile based object recognition. A limitation of monomodal tactile based recognition would be the inherent sample inefficiency of the tactile actions and lots of probing actions are required to collect data for recognition tasks. Intelligent combination with visual perception can greatly improve the efficiency and robustness of the task.

The techniques for visuo-tactile sensory information fusion needs to be discussed. Typically multi-modal fusion is performed with (a) data-level fusion, (b) feature-level fusion and (c) decision-level fusion as illustrated in Fig. 2.4. In *data-level fusion*, heterogeneous sensory data are standardized to a uniform format and integrated through any sort of union procedure. One such method is expressing vision data from RGB-D images and tactile data from point-contact and array-based sensors as a set of points and use point cloud techniques for object recognition. Gandler et al. (2020) used such techniques for combining visuo-tactile point clouds at the data level to perform recognition and surface reconstruction tasks. Similar works have been presented point cloud level fusion for visuo-tactile data (Smith et al., 2020; Björkman et al., 2013). *Feature-level fusion* involves independently extracting a specific number of features from visual and tactile data, and subsequently merging these features. This approach is commonly employed to prevent information disparity across various formats, allowing the features to be seamlessly integrated into different machine learning models. Liu et al. (2016b) introduced the problem of *weak pairing* between tactile and vision domains i.e., since visual data and tactile data are usually collected separately, there is no one-to-one correspondence between the two domains, instead a group of samples from the vision modality corresponds to a group of samples in the tactile modality. The authors perform early fusion of the multi-modal data by using multivariate-time-series model to represent the tactile sequence and a covariance descriptor to characterize the image. They propose a joint group kernel sparse coding (JGKSC) method to tackle the intrinsically weak pairing problem and employ it to recognise house-

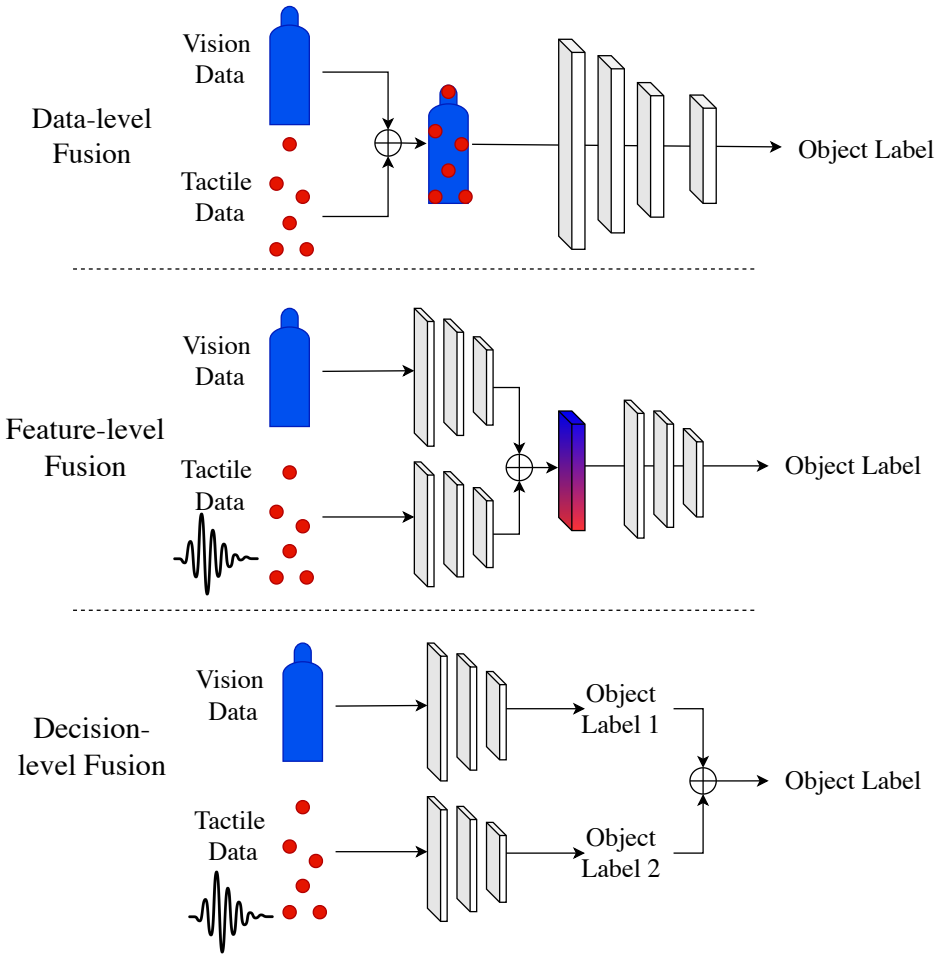


Figure 2.4: Common visuo-tactile fusion techniques for object recognition

hold objects. Luo et al. (2015) employed identical feature extraction methods for visual images and tactile images in order to recognise and localise objects. Abderrahmane et al. (2018) propose a *zero-shot* object recognition framework using vision and tactile data to recognise daily object that have never been seen or touched before. Raw data from the BioTac sensor are used for tactile data and RGB images for vision data and dimensionality reduction is performed to reduce the number of signals from the raw BioTac data. Separate CNN are used for feature extraction which is then combined to a feature vector in order to learn attributes for zero-shot object recognition. Kroemer et al. (2011) tackle the problem of material classification using touch. Since tactile data is high dimensional and noisy, the authors propose a method to learn a lower dimensional representation of tactile using vision data of the surfaces, thereby working with the weak-pairing issue of vision and tactile. The experiments demonstrate high performance using tactile sensing alone for ma-

terial recognition. *Decision-level fusion* methods handles the vision and tactile information independently, extracts features to make decisions, and finally combines or evaluates these decisions. For instance, Corradi et al. (2017) extracted the features from visual and tactile images independently and the decisions for object classification were based on weighted average of the posterior estimations.

While aforementioned works focused on multimodal sensor fusion, transfer of perceptual information from one modality to the other, termed cross-modal perception, is gaining research attention in recent years. Cross-modal perception enables the robots to switch to another sensing modality in case of sensor failure/ unavailability from one sensing modality in order to complete the task. Such methods leverage the pre-trained models available from one modality to train the other modality thereby reducing the need for data annotation, training and so on. Falco et al. (2019) proposed a cross-modal learning method between vision and tactile for an object recognition task. Point cloud data was acquired using a RGB-D sensor and custom array-based tactile skin on the robot end-effector. The point clouds from the two domains were equalised for partiality and for point density and hand-crafted feature descriptors were used to extract features from both visual and tactile point clouds. Subsequently, a support vector machine (SVM) based model was trained on visual features and during execution stage the robot leverages the trained model to recognise objects using tactile perception. Similarly to point clouds, array-based tactile sensor data can be converted to grey-scale images or heatmaps through normalization and thresholding. They can be used with images captured from visual sensors using computer vision methods for object recognition (Lee et al., 2019a). Recently, Purri and Dana (2020) tackle the interesting problem of vision to tactile cross modal transfer in order to *infer* tactile data from only vision data alone of sample object surfaces. Although an extremely hard task for robots, the results are encouraging to inspire research in the direction of synesthesia (the production of an experience relating to one sense by a stimulation of another sense) or vision-to-tactile transfer and vice-versa. Similarly, Takahashi and Tan (2019) also proposed a encoder-decoder deep learning network with latent variables to estimate tactile data from images. Tactile data is recorded by stroking object surfaces during training alone and is estimated using visual images during testing. However, the authors use restrictive assumptions such as ignoring the initial samples of stroking which produces spurious data due to overcoming static friction, this however can be crucial in order to distinguish object properties. On the other hand, Zheng et al. (2019) presented a method for tactile-to-vision transfer by which a novel object is recognised by using its tactile signal to retrieve perceptually similar surface material images through the learnt cross-modal correlation.

Similarly for object recognition, surface reconstruction tasks have also been explored

with visuo-tactile sensing. Gaussian process implicit surface (GPIS) has been widely used for tactile-based object reconstruction (Dragiev et al., 2011; Yi et al., 2016; Björkman et al., 2013; Gandler et al., 2020; Martens et al., 2016; Suresh et al., 2021; Jamali et al., 2016). The implicit surface described by a Gaussian process describes the shape of an object through a function that decides for each point in space whether it is part of the object or not. It produces smooth surface manifolds with a reasonable number of tactile points as input and also provides probabilistic information to guide the tactile actions. However, for complex shapes it typically requires lots of points uniformly distributed on the object's surface for reconstruction (Jamali et al., 2016). Certain studies have integrated tactile sensing with visual perception to augment shape completion by leveraging prior information acquired through visual cameras (Gandler et al., 2020; Smith et al., 2020). Similarly, Wang et al. (2018) designed a framework for generating 3D shape of objects from a single visual image using learnt shape priors which are refined using tactile sensing.

2.4.1 Limitations in the state-of-the-art

Prior works in cross-modal visuo-tactile object recognition tackled the weak-pairing problem between the visual and tactile data by explicitly equalising the point density and predefined hand-crafted features that limit the applicability to real world scenarios (Falco et al., 2019). In Chap. 7, these issues are tackled by proposing a novel cross-modal domain adaptation method that is capable of working with raw dense visual and sparse tactile point clouds directly for the object recognition task. Considering object reconstruction, popular techniques such as GPIS fail to capture fine shape details with sparse tactile input data (Jamali et al., 2016; Björkman et al., 2013; Gandler et al., 2020). Furthermore, directly deploying deep-learning based strategies for shape completion with sparse input data is impractical as the collection of a large dataset of tactile data for training is prohibitively expensive. Moreover, prior works employed touch as a method to refine the shape estimate from vision in two-step sequential process that is usually time consuming (Gandler et al., 2020; Smith et al., 2020). In this thesis, these open challenges are tackled by proposing a novel deep learning method for tactile-based reconstruction of transparent objects leveraging large-scale synthetic datasets described in Chap. 8. The reconstruction method is also deployed for visuo-tactile data which are collected using a shared perception method with a joint information gain approach to improve the sample efficiency of actions for data collection in Chap. 5.

2.5 Visuo-Tactile based Interactive Perception

Interactive Perception or *perceptive manipulation* is any kind of purposeful manipulation actions performed to simplify or enhance the perception of the environment (Bohg et al., 2017). Interactive perception techniques rely upon effective scene understanding in order to plan and execute manipulative actions. The scene understanding iteratively improves upon performing manipulation actions. In unstructured cluttered scenarios, the target object may have multiple other objects overlapping on it in random configurations. A typical choice for scene understanding in computer vision is *scene graphs* which is a data structure that describes objects in a scene and the relationships between these objects (Johnson et al., 2015). Support graphs, a type of scene graph have been introduced to describe the support relations between objects in the scene through geometric reasoning (Kartmann et al., 2018; Mojtahedzadeh et al., 2015; Schwarz et al., 2018). Sui et al. (2017) presented an axiomatic scene estimation method to describe the relationship between objects and object poses as a scene graph for manipulation. Mitash et al. (2022) developed a Monte Carlo Tree Search-based technique for scene understanding leveraging physics-priors of objects in clutter for pose estimation. Zhang et al. (2021) tackled the issue of inferring object relationships through a neural network performing classification task on all possible pairwise permutations between the objects in the scene. Scene understanding is followed by planning manipulation actions in clutter which is a challenging task and has received immense research interest. Prehensile manipulation such as grasping and non-prehensile manipulation such as pushing are frequently used for interactive perception and detailed in the following subsections.

2.5.1 Visuo-Tactile based Prehensile Manipulation

Prehensile manipulation, or grasping can be approached via model based/analytical or data-driven methods (Bohg et al., 2013). Analytical methods are based on force analysis and rigid body dynamics. They are employed for objects or known geometry, mass and friction properties and are widely used in controlled environments for instance automotive production lines. However, analytical methods perform poorly in unstructured or dynamic environments, with real world objects with varying friction coefficients and deformability and so on. Model misspecifications and unmodelled model factors contribute to the reduction in performance (Bohg et al., 2013). Some works on analytical methods for grasping can be found in (Kim et al., 2013; Rodriguez et al., 2012; Horowitz and Burdick, 2012; Kazemi et al., 2012; Gopalakrishnan and Goldberg, 2005). Alternatively, with data-driven methods, grasp configurations are statistically learnt. These can be from 3D mesh mod-

els, point clouds, or images, from which geometry can be used to infer grasp stability, and appearance can be used for detection of areas suitable for grasping. Grasp outcomes can be learnt from human supervision (Kamon et al., 1996; Lenz et al., 2015), from simulation (Kappler et al., 2015; Johns et al., 2016; Mahler et al., 2016) or from robot data collection (Pinto and Gupta, 2016; Levine et al., 2018). Typically, data-driven methods rely on geometry and vision sensing and reason on grasp outcomes before actual contact while tactile data and contact forces are not taken into consideration. A detailed review on data-driven methods for grasping is provided by Bohg et al. (2013). This section focusses on techniques wherein vision and tactile sensing together are used for grasping.

Typically, visual input is used for grasp generation/planning and tactile sensing is used for closed loop control once in contact with the object. Son et al. (1996) performed experiments by using visual feedback to perform rough positioning of the hand and tactile feedback to detect contact, in order to compensate for the difference between the orientation of the object and the gripper and to control grasp force for delicate manipulation tasks. They compare with force sensor based approaches and empirically show that visuo-tactile approach provides a more gentle grasp. It is important to note that techniques provided in Sec. 2.3 on visuo-tactile object pose estimation are typically performed before grasping tasks. Kragic et al. (2003) compensated for imperfect vision-based pose estimation in order to center the object with the gripper using tactile sensing. Morales et al. (2007) demonstrated an experiment combining analytical grasp model primitives with vision and tactile sensing in order to remove a book from a bookshelf. Li et al. (2015) proposed a control framework that unifies visual and tactile servoing for robust manipulation. Allen and Michelman (1990) used visual sensing and tactile sensing to calculate the position of contact along a finger in order to estimate applied finger forces for grasping tasks.

Considering data-driven visuo-tactile grasping models, Calandra et al. (2018) proposed an end-to-end action-conditional model that iteratively adjusts a robot's grasp based on raw visuo-tactile inputs. The authors use tactile images from the GelSight sensor, RGB images and actions and then employ a late fusion approach by extracting information using tactile CNN, image CNN and a two-layer fully-connected perceptron respectively. They empirically show that the proposed approach outperforms a variety of baselines at estimating grasp outcomes and provide interpretable re-grasping behaviours. Recently, Kumar et al. (2019) demonstrated a sim-to-real approach to train reinforcement learning policies using vision and tactile sensing on gripper fingertips. Chebotar et al. (2016) proposed a self-supervision framework for regrasping on tactile data alone. Initially, a grasp stability predictor is designed that uses spatio-temporal tactile features collected from the early-object-lifting phase to predict the grasp outcome with a high accuracy. The trained predic-

tor is then used to supervise and provide feedback to a reinforcement learning algorithm that learns the required grasp adjustments based on tactile feedback. Cui et al. (2020a) predicted the success of grasps using visuo-tactile fusion based on *Self-Attention* Mechanism. Furthermore, the authors in another work propose a 3D CNNs based visual-tactile fusion network to assess grasp states of deformable objects (Cui et al., 2020b). Lee et al. (2019b) used self-supervision to learn a compact and multimodal representation of vision, force/-torque, and proprioception, which can be used to improve the sample efficiency of policy learning. For grasping of unknown objects, prior works rely upon global shape or features from sensory data and a set of heuristics (Bohg et al., 2013; Schaub and Schöttl, 2020; Morrison et al., 2020; Schmidt et al., 2018). For instance, Morrison et al. (2020) developed an object-independent grasp synthesis method from depth images using their generative grasping convolutional neural network (GG-CNN). Fazeli et al. (2019) proposed a method to emulate hierarchical reasoning and multi-sensory fusion in a robot that learns to play Jenga, a complex game that requires physical interaction to be played expertly. This model captures latent descriptive structures, and the robot learns probabilistic models of these relationships in force and visual domains through a short exploration phase.

Frequently, the task of robotic grasping of unknown and complex objects presents significant challenges, which are exacerbated when these objects are situated within cluttered environments. Under these circumstances, the application of non-prehensile manipulation techniques becomes pertinent.

2.5.2 Visuo-Tactile based Non-Prehensile Manipulation

Non-prehensile manipulation, meaning manipulation without grasping, is useful when it comes to interactive perception (Bohg et al., 2017). Non-prehensile manipulation includes actions such as pushing, throwing, flipping, and so on. Similar to prehensile manipulation, existing techniques for non-prehensile manipulation techniques fall into two categories: model-based and data-driven methods (Stüber et al., 2020). The first type relies on analytical, physics-based models of pusher object interactions within higher-level planning and control frameworks (Stüber et al., 2020). The second type uses data-driven methods to construct forward, or inverse models of pusher-object interactions, or to directly learn a pushing control policy (Lloyd and Lepora, 2020). Multiple analytical model-based techniques have been proposed in the literature. Mason (1986a) proposed the voting theorem for determining the direction of rotation of a pushed object. This rule depends only on the centre of mass of an object and not the underlying distribution of support forces. Lynch et al. (1992) used an optical waveguide tactile sensor mounted on one finger of a gripper attached to achieve stable translation of a rectangular object and circular disk on a

moving conveyer belt. In (Jia and Erdmann, 1999), the authors showed that it is possible to determine pose and motion of a planar known object solely from tactile sensing feedback by pushing. Analytical models are based on well-understood physics models but employ restrictive assumptions and assumptions that do not generalise well in practice. For example, many of them assume homogeneous, isotropic and stationary friction, which may not be valid for some surface materials (Yu et al., 2016).

Considering data-driven methods Meier et al. (2016) used 2D tactile images in order to learn a CNN classifier to distinguish between two types of sliding and slipping while performing a pushing operation. Hellman (2016) used BioTac sensor data for a task of closing a deformable ziplock bag and used the data to learn a reinforcement learning policy for contour following. Vision alone in the context of non-prehensile manipulation has been used to predict the state of the object after manipulation. For instance, Agrawal et al. (2016) used a Siamese CNN in order to learn the poking location, angle and length to move an object from one location to another location. Similarly, Finn and Levine (2017) learned to predict the result image after pushing an object. This was used for planning pushing actions in order to satisfy the goal states provided by the user. Li et al. (2018) proposed Push-net, a deep recurrent neural network (RNN) to tackle the problem of quasi-static planar pushing to re-orient and re-position objects. Their approach requires only visual camera images as input and remembers pushing interactions using a long short-term memory (LSTM) module. However, with vision-alone non-prehensile manipulation tasks have shown impressive results, the tasks are inherently primitive. In (Bauza et al., 2018), the authors use Gaussian processes to model the change in position and orientation of an object in response to a push at a specified contact position and angle, and embed the model in a Model Predictive Control (MPC) framework. Attempting to combine the data-driven and analytical methods, Zhou et al. (2018a) proposed a convex polynomial model for planar sliding. The authors approximate the limit surface using a simple parameterized model (the level set of a convex polynomial) and fit the model using a computationally efficient identification procedure. Reinforcement learning (RL) has emerged as a promising method for non-prehensile manipulation (Lowrey et al., 2018; Clavera et al., 2017). In Clavera et al. (2017), the authors proposed to decompose the system into modules to train in simulation instead of end-to-end training. They demonstrate reliable to sim-to-real transfer and the robot capable of handling the pushing of objects from different initial and final states. Lloyd and Lepora (2020) designed a reactive and adaptive method for robotic pushing that uses rich feedback from a high-resolution optical tactile sensor to control push movements instead of relying on analytical or data-driven models of push interactions. More specifically, they use goal-driven tactile exploration to actively search for stable pushing configurations that cause the

object to maintain its pose relative to the pusher while incrementally moving the pusher and object towards the target. They tested the framework on planar and curved surfaces. A detailed review on the state-of-the-art of non-prehensile manipulation can be found in (Stüber et al., 2020).

2.5.3 Limitations in the state-of-the-art

Most approaches for grasping in literature are focused on singulated objects or objects in sparse clutter without overlapping or occluding objects as they rely primarily upon visual perception (Calandra et al., 2018; Cui et al., 2020b; Morales et al., 2007). Objects in dense clutter necessitates the use of interactive perception and mechanical search techniques where the objects need to be rearranged autonomously to retrieve a desired target object (Danielczuk et al., 2019; Bohg et al., 2017). Furthermore, most works either employ grasp or push actions for manipulation (Zeng et al., 2018; Danielczuk et al., 2019). Methodologies that allow the robot to choose to perform prehensile or non-prehensile manipulation based on the type of object would be very beneficial and is absent in current research. Moreover, most works in this domain target the manipulation of rigid objects. However, objects may have articulation joints such as drawers and glasses which cannot be manipulated in the same manner (Bohg et al., 2013). This thesis tackles these challenges by developing a visuo-tactile based interactive perception method to autonomously declutter a densely cluttered scene using both prehensile and non-prehensile actions based on object type as presented in Chap. 5. The goal-driven manipulation of articulated objects, a challenging and open problem in literature, is tackled with the use of both types of manipulation actions as detailed in Chap. 6.

Chapter 3

System Description

In this chapter, the experimental system including robots and sensors that have been used in this thesis has been described.

3.1 Robotic System

The system consists of two robots: UR5 robot from Universal Robots with the Robotiq 2F140 gripper and Panda robot from Franka Emika with the standard gripper from the manufacturer as shown in Fig. 3.1. The technical specifications of the robots are shown in the Tab. 3.1. As seen from the Tab. 3.1, both robots have similar reach, weight and joint speed and repeatability. The payload capability of the UR5 was higher than the Panda

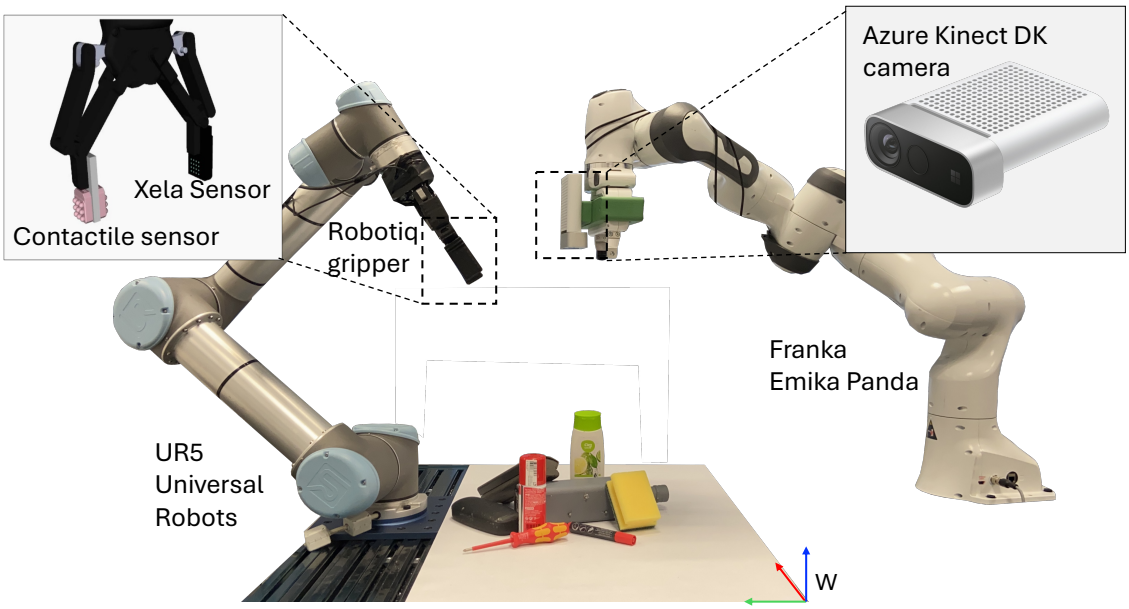


Figure 3.1: Robot and sensor system description

robot. The Panda robot offered 7 degrees-of-freedom from the 7 rotational joints while the UR5 robot had 6 rotational joints. The Panda robot also had force/ torque sensors embedded on all seven joints. Since the UR5 robot has a higher payload, and the Robotiq gripper provided easy reconfigurability, it had been sensorised with the tactile sensors on the fingertips. The Panda robot with the additional degree-of-freedom allowed fine-grained control for reaching various positions in the joint workspace of the robot and was sensorised with the vision sensor. Since the camera was attached to the end-effector of the Panda robot, the setup mimicked human-like system wherein the eyes (camera) is on a movable head/ neck (Panda robot) and the hand mimicked the UR5 robot with tactile sensing. Apart from certain manipulation actions in Chap. 6 where the Panda's torque sensors were used for interaction with the objects, in all experiments in this thesis, the UR5 robot with tactile sensors was primarily used for robot-object interactions whereas the Panda robot was used for capturing images from various viewpoints without contact with the objects. Furthermore, as seen from Fig. 3.1, the robots were attached to a custom designed base frame which forms the shared workspace of the robots where the objects have been placed. The workspace size was (1 m, 0.55 m) and a common world coordinate frame W is placed at the corner of the workspace. The maximum allowed speeds for the UR5 and Panda were 75 mm/s and 100 mm/s respectively for safety constraints.

3.2 Sensor System

3.2.1 Vision Sensor

The Azure Kinect DK sensor from Microsoft©has been used as the vision sensor. It has a 1 MegaPixel(MP) depth camera, 12 MP red-green-blue (RGB) camera, microphone array and inertial measurement unit (IMU) for motion sensing. The RGB camera was used with the following settings: image resolution 2048×1536 pixels and 15 frames-per-

Table 3.1: Technical specifications of the robots

Parameters	**UR5 robot**	**Panda robot**
DoF	6	7
Payload	5kg	3kg
Reach	850mm	855mm
Repeatability	$\pm$0.1mm	$\pm$0.1mm
Weight	18.4kg	18kg
Speed	180°/s	150-180°/s
Sensors	-	Torque sensors

second (fps). The depth camera was used with the following settings: image resolution 1024×1024, 15fps and operating range 0.25-2.21m. The camera synchronised provided RGB and depth (RGB-D) images and corresponding coloured point cloud in $\{x, y, z, r, g, b\}$ format. The microphone array and IMU sensors were not used in any experiment in this thesis. The intrinsics of the camera was calibrated using the checkerboard approach (OpenCV, 2024) and the extrinsic calibration of the camera was calculated using hand-eye calibration techniques (more information is provided in Chap. 5). The point clouds were expressed in the common world coordinate frame of the system using the camera extrinsics. The Azure Kinect ROS driver (Microsoft, 2024) was used to acquire the images and point clouds from the camera and pipe through to all nodes using the data.

3.2.2 Tactile Sensors

As detailed in Sec. 2.2, there is no standard tactile sensor for robotic applications. There are various types of sensors based on the transduction mechanism to measure the external pressure/ force. This introduces a challenge for comparing various types of tactile sensors and associated algorithms. For instance, algorithms using vision-based tactile sensors can employ computer vision techniques usually associated with camera sensors whereas these methods cannot be transferred to other types of tactile sensors such as capacitive or magnetic-based. In order to ensure the developed methods in this thesis to be agnostic to the type of tactile sensor, only 3D contact information in terms of point clouds based on contact forces were used which can be provided by all types of tactile sensors. Furthermore, two different types of off-the-shelf tactile sensors have been used in this thesis: the Contactile sensor and the Xela tactile sensor. These sensors have different operating principles and were used to demonstrate the applicability of the proposed methods to various types of tactile sensors.

Contactile Sensor

The Contactile sensor consists of multiple pillar-like devices arranged in a 3×3 array called as the PapillArray (Khamis et al., 2019). The principle of the contactile sensor is shown in Fig. 3.2. The sensors works on the pinhole camera principle where a pinhole aperture is created in a printed circuit board and a diffuse reflector is embedded inside the PapillArray pillar (Fig. 3.2b). The reflector is illuminated by two LEDs either side of the pinhole aperture (Fig. 3.2c). The reflected light forms an inverted image on a quadrant photodiode (four separate photodiodes in a segmented configuration). The projected image forms a spot of light whose shape, position, and size depends on the 3D displacement of the reflector disk. A multivariate polynomial regression algorithm was used to infer the external force applied

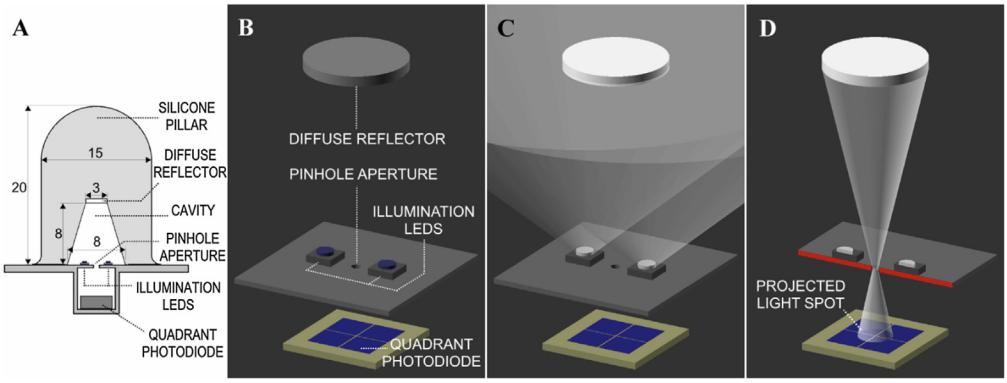


Figure 3.2: Principle of contactile sensor: a) Cross-section of a pillar of the sensor. Dimensions shown in mm. b) 3D rendered illustration of reflector, LEDs, pinhole aperture, and quadrant photodiode sensor. c) Infrared LEDs flood the truncated-conical pillar cavity with light causing the diffuse reflector at the top of the cavity to behave as a light source. d) Light from the reflector passes through the aperture and a light spot is projected below. Reproduced with permission from Khamis et al. (2019)

.

based on the displacement of the reflector disk and calibration was performed to provide 3-axis forces (in Newtons) and pillar displacement measurements. The sensor's provided various possible sampling frequencies between 100-1000Hz. The dimensions of a 3×3 array was $24.0 \times 30.6 \times 12.8$mm and spatial density was 7mm between the pillars. The force resolution was 0.05N. Due to the relative hard silicone pillar, the hysteresis was negligible but the sensitivity was lower than the Xela tactile sensor. As each pillar provided independent measurements, the sensor was also capable of detecting incipient slipping, which can be very useful for precise grasping manoeuvres. In this thesis, two 3×3 Contactile sensors were used and placed on the outer and inner sides of one fingertip of the Robotiq 2F140 gripper as shown in Fig. 3.1.

Xela Tactile Sensor

The Xela tactile sensors were procured from XELA robotics. These were array based tactile sensors available in various matrix configurations: 4×4, 4×6 and 6×1. The sensor's working principle is shown in Fig. 3.3. It has three layers: the outer shell made of fabric which offers protection, the elastomer layer composed of multiple magnets in an array configuration and the MLX90393 chip which is a magnetometer that recognises the displacement of the magnet in 3-dimensions. On application of external force, the magnets can move in x, y and z directions which results in magnetic field changes. The magnetic field changes are converted to forces measurements. To reduce cross-talk effect due to

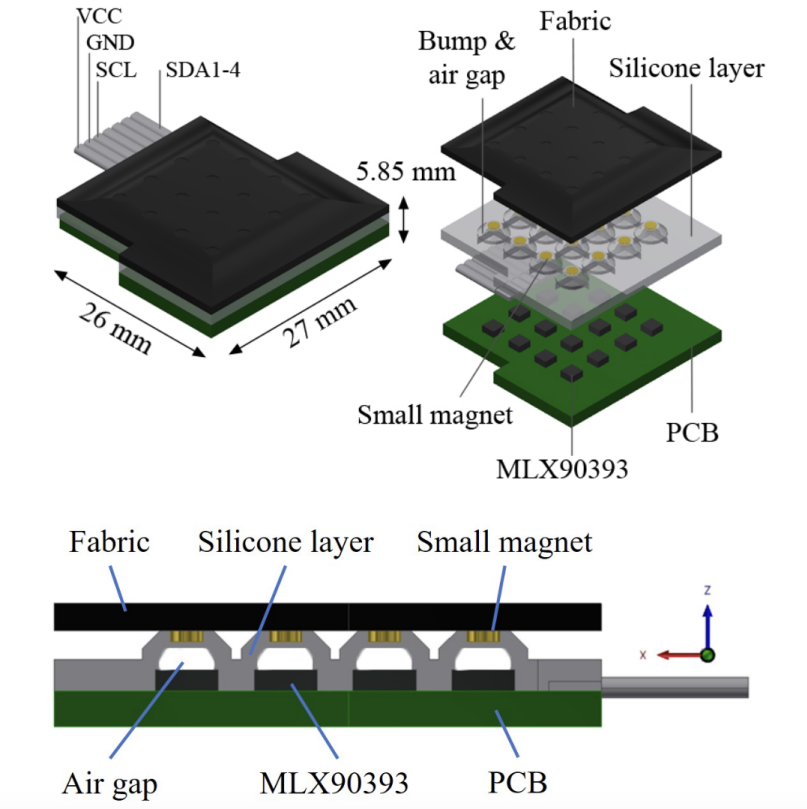


Figure 3.3: Xela tactile sensor. The sensor functions on hall-effect principle. Reproduced with permission from Tomo et al. (2018) ©IEEE

.

multiple magnets in close proximity, an air-gap was introduced inside the elastomer silicon layer which acts as a deformable spring-like structure. The magnet and the corresponding magnetometer sensor together forms one sensing unit called as a taxel. The distance between each taxel was 4.7mm and the 16 taxels were designed for a 4×4 configuration with sensor size $26 \times 27 \times 6.05$mm. The sampling frequency of the sensor was 100Hz. The sensor's output values were not calibrated to force (Newtons) but rather provided raw values in the range of 36000 - 45000 which were normalised. The Xela tactile sensors were attached to one of the fingers of the Robotiq 2F140 gripper. The sensors provided a sensitivity of 0.001N according to the datasheet (Tomo, 2019). The sensors have a built-in temperature compensation mechanism as well (Tomo, 2019). Due to the flexible fabric covering, the sensor suffers from hysteresis. During continuous experiments, the robot was periodically commanded to wait between data acquisitions after every 10 probing interactions which allowed to compensate for hysteresis and recalibrate the baseline value. Furthermore, since the sensor was based on Hall-effect principle, it was applicable to non-ferromagnetic ob-

jects only. For objects that may contain ferromagnetic material, the Contactile sensor was used instead. The 6×4 sensor array was placed on the outer side, 4×4 array on the inner side and 6×1 array on the bottom side of one finger of the Robotiq 2F140 gripper as shown in Fig. 3.1.

3.3 Miscellaneous Hardware and Software

The robot experiments in the thesis were performed on a workstation using Ubuntu 18.04 with Intel©Xeon Gold 5222 CPU and 16GB RAM. For training neural networks, a Nvidia©Quadro RTX 4000 GPU was used. For ground-truth annotation of the pose of objects in Chap. 4-6, the OptiTrack©PrimeX22 cameras were used. The OptiTrack cameras use markers to track pose of objects within the a predefined volume. Six OptiTrack PrimeX22 cameras were used, which provide 0.1mm accuracy and 360 frames-per-second frame rate. The Motive software (NaturalPoint, Inc.) was used for data extraction from the OptiTrack cameras. Ethernet communication was used to connect the robots and the OptiTrack cameras. The USB data interface was used for the Azure Kinect DK camera, the Contactile sensor, and the Xela Tactile sensor (through CAN-to-USB driver). Robot Operating System (ROS) Melodic (Quigley et al., 2009) middleware provided the hardware drivers for the robots and sensors. The MoveIt motion planning framework (Chitta, 2016) was used for trajectory planning and control of the robots. Point cloud processing was performed using Point Cloud Library (PCL) (Rusu and Cousins, 2011) and Open3D (Zhou et al., 2018c). The main codebase was built in C++ for Chap. 4-6 and Python was used for neural network architectures in Chap. 7-8. The experimental objects consisted of everyday household objects such as bottles, mugs, cans and so on which were intentionally chosen for ease of reproduction. The summary of the main hardware has been summarised in the Tab. 3.2.

Table 3.2: Summary of the hardware used in the thesis

Component	**Quantity**
UR5 Universal Robots	1
Franka Emika Panda	1
Robotiq 2F140 Gripper	1
Microsoft Azure Kinect DK	1
Contactile sensor 3×3 array	2
Xela Tactile Sensor 4×4 array	1
Xela Tactile Sensor 6×4 array	1
Xela Tactile Sensor 6×1 array	1
OptiTrack PrimeX22 cameras	6

Part II

Shared Visuo-Tactile Interactive Perception for Object Pose Estimation

Chapter 4

TIQF: Translation-Invariant Quaternion Filter for Visuo-Tactile based Pose Estimation

Parts of this chapter are **published** as:

- "*Active Visuo-Tactile Point Cloud Registration for Accurate Pose Estimation of Objects in an Unknown Workspace*," **P. K. Murali**, M. Gentner, and M. Kaboli in 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2021, pp. 2838-2844 (Murali et al., 2021)
- "*An Empirical Evaluation of Various Information Gain Criteria for Active Tactile Action Selection for Pose Estimation*," The IEEE International Conference on Flexible and Printable Sensors and Systems (FLEPS 2022), pp. 1–4 in **P.K. Murali**, R. Dahiya, and M. Kaboli (Murali et al., 2022a)

The video of the experiments from this chapter is available here:
`https://drive.google.com/file/d/1ud7Vc5LkjG7HCuIvnURwaKKEy-K5Ri-s/view?usp=sharing`

4.1 Introduction

Accurate estimation of object pose (translation and rotation) is crucial for autonomous robots to grasp and manipulate objects in an unstructured environment. Even small inaccuracies in the belief of the object pose can generate incorrect grasp configurations and lead to failures in manipulation tasks (Li et al., 2020a). Vision sensor-based strategies

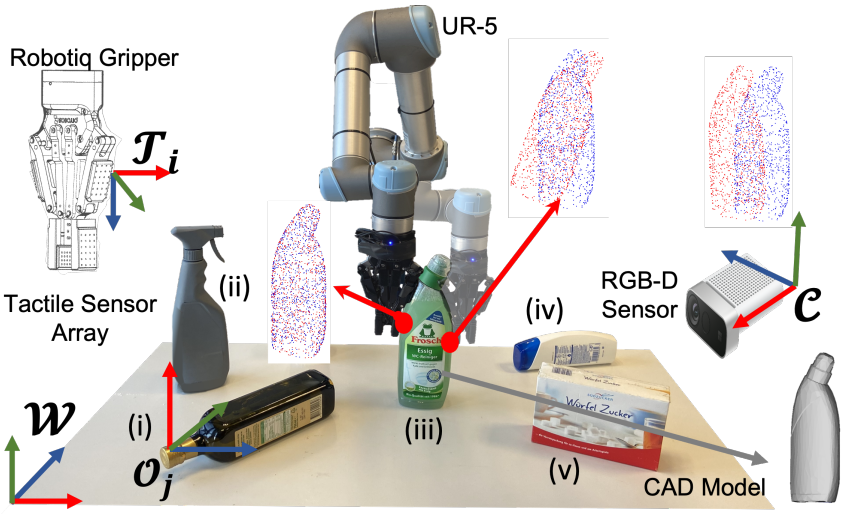


Figure 4.1: Experimental setup. A Robotiq two-finger adaptive robot gripper is equipped with 3-axis tactile sensor arrays and mounted on a UR5 robotic arm. In this figure, 6 experimental objects are selected and placed in the workspace. The experimental objects constitute daily objects as follows: (i) olive oil bottle, (ii) spray, (iii) cleaner, (iv) shampoo, (v) sugar box. In experiments, objects were placed in the workspace with various locations and orientations.

are prevalently employed for object pose estimation; nevertheless, the pose estimates are often afflicted by residual uncertainty arising from improper sensor calibration, adverse environmental conditions (such as occlusions, extreme illumination, and poor visibility), and inherent object characteristics (including transparency, specularity, and reflectivity). Tactile sensors in combination with robot proprioception provides high fidelity local measurements regarding object pose. However, mapping entire objects using tactile sensors is highly inefficient and time-consuming which necessitates the use of intelligent data gathering strategies and combining vision sensing to drive the tactile sensing (Hsiao et al., 2011).

This chapter presents a *novel* framework for active visuo-tactile point cloud registration for accurate pose estimation of objects. The contributions are as follows:

(I) A novel formulation termed translation-invariant quaternion filter (TIQF) for dense vision-based point clouds and sparse tactile-based point clouds for point cloud registration.

(II) An active touch strategy to enable the robot to generate candidate actions and select the optimal action strategically based on information gain. The active strategy is demonstrated to be computationally efficient to perform an exhaustive uncertainty-based action selection in real-time without the need for trading information gain with execution time. The vision pose estimate is corrected by using the tactile modality

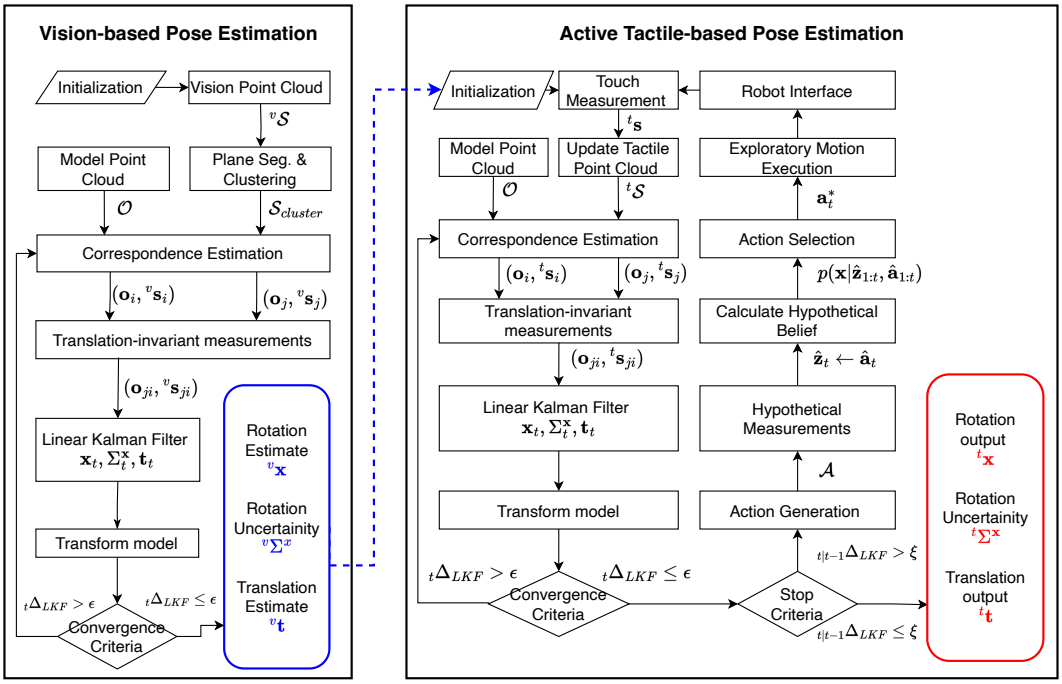


Figure 4.2: The proposed framework for an active visuo-tactile point cloud registration for the accurate object localization

using the active touch strategy.

(III) Extensive experiments have been performed on benchmark datasets and robotic setup shown in Fig. 4.1 to compare the proposed approach against the random exploration strategy.

4.2 Methodology

4.2.1 Problem Formulation

An active visuo-tactile based pose estimation framework is proposed, as shown in Fig. 4.2. The problem is formally defined as follows: given N_O objects with designated frames $\mathcal{F}_k$ with $k = 1, \ldots, N_O$ in the workspace W_{XYZ} of the robot with unknown poses. The workspace W_{XYZ} is defined as a discretised 3D grid bounded by the kinematic reaching constraints of the robot and defined in the world coordinate frame $\mathcal{W}$. The pose of each object is presumed to remain invariant over time. The objective is to find the object pose ${}^{\mathcal{W}}H_{\mathcal{F}_k}$ given sensor measurements ${}^{v}\mathcal{S}$ from vision sensor and tactile sensor ${}^{t}\mathcal{S}$. The proposed approach is used to verify and correct the pose of the object obtained from visual sensing with tactile perception.

4.2.2 Proposed Framework

As described in the framework in Fig. 4.2, vision-based pose estimation is used for providing an initial estimate which is subsequently corrected by active tactile-based estimation. In Fig. 4.2a, the point cloud ${}^{v}\mathcal{S}$ is captured by the vision sensor and is transformed to the world frame $\mathcal{W}$ by applying the ${}^{\mathcal{W}}H_{\mathcal{C}}$ homogeneous transformation typically known as the hand-eye transformation, where $\mathcal{C}$ is the camera frame. Pre-processing steps such as plane segmentation and clustering are performed to extract the points corresponding to the object of interest ($\mathcal{S}_{cluster}$). Subsequently, the translation-invariant quaternion filter (TIQF) approach which is a probabilistic pose estimation method, is used for pose estimation. TIQF works on a putative set of correspondences between the model and scene clouds which is found in the correspondence estimation step using the closest point rule (Besl and McKay, 1992). The TIQF algorithm upon convergence provides the rotation estimate ${}^{v}\mathbf{x}$, the corresponding rotation uncertainty ${}^{v}\mathbf{\Sigma}^{\mathbf{x}}$ and translation estimate ${}^{v}\mathbf{t}$. The pose estimate from vision is used in order to initialise the active tactile-based pose estimation procedure as there can be residual errors in pose estimation from vision-based sensors that can be corrected with high-fidelity tactile measurements. The tactile-based pose estimation shown in Fig. 4.2b is also performed using the TIQF algorithm and shows the adaptability of the algorithm to handle batch data and sequential data as well as dense and sparse data. Tactile point cloud data are collected sequentially forming the tactile point cloud ${}^{t}\mathcal{S}$. TIQF is performed to get a rotation and translation estimate ($\mathbf{x}, \mathbf{t}$ respectively). Furthermore, an active touch selection strategy is presented as described in Sec. 4.2.4 in order to intelligently and efficiently extract tactile measurements (a_t^* as shown in Fig. 4.2b), as performing each tactile measurement is a time-consuming process. The TIQF method and the active tactile exploration approach are detailed in the subsequent sections.

4.2.3 Translation-Invariant Quaternion Filter (TIQF)

To solve the point cloud registration problem for the vision-based pose estimation and the active tactile-based pose estimation in the same manner, a method termed translation-invariant Quaternion filter (TIQF) is presented. For point clouds from a vision sensor, the TIQF algorithm can be used in a batch manner and during active tactile exploration it can handle sequential point measurements as well. The point cloud registration problem given known correspondences can be formalised as follows:

$$\mathbf{s}_i = \mathbf{S}.\mathbf{Ro}_i + \mathbf{t} \quad i = 1, \ldots N \tag{4.1}$$

where $\mathbf{s}_i \in \mathbb{R}^3$ are points belonging to the scene cloud $\mathcal{S}$ drawn from sensor measurements and $\mathbf{o}_i \in \mathbb{R}^3$ are the corresponding points belonging to the model cloud $\mathcal{O}$. The scale, rotation and translation are defined as $\mathbf{S} \in \mathbb{R}^3$, $\mathbf{R} \in SO(3)$ and $\mathbf{t} \in \mathbb{R}^3$ respectively which are unknown and need to be computed in order to align $\mathbf{o}_i$ with $\mathbf{s}_i$. In the case where the object model is known a priori $\mathcal{O}$, the scale parameter is set to 1 i.e., $\mathbf{S} = \{1,1,1\}$. The rotation and translation estimation are decoupled, and the translation estimate can be trivially computed once rotation is known. Given a pair of correspondences $(\mathbf{s}_i, \mathbf{o}_i)$ and $(\mathbf{s}_j, \mathbf{o}_j)$, the following can be defined $\mathbf{s}_{ji} = \mathbf{s}_j - \mathbf{s}_i$ and $\mathbf{o}_{ji} = \mathbf{o}_j - \mathbf{o}_i$. From Eq. (4.1):

$$\mathbf{s}_j - \mathbf{s}_i = (\mathbf{R}\mathbf{o}_j + \mathbf{t}) - (\mathbf{R}\mathbf{o}_i + \mathbf{t}) \tag{4.2}$$

$$\mathbf{s}_{ji} = \mathbf{R}\mathbf{o}_{ji} \tag{4.3}$$

Eq. (4.3) is independent of $\mathbf{t}$ and once rotation $\mathbf{R}$ is estimated, the translation $\mathbf{t}$ can be obtained in closed form from Eq. (4.1).

The rotation estimation problem is cast into a Bayesian estimation framework. The rotation estimate $\mathbf{R}$ is denoted in its quaternion form as the state $\mathbf{x}$ which needs to be estimated by measurements $\mathbf{z}$ obtained via actions $\mathbf{a}$ upto time t. The various methods of representing rotations are provided in the Appendix A. Quaternions were chosen as (a) they do not suffer from Gimbal lock issues present in Euler angles, (b) faster computation and memory efficient compared to rotation matrices as 4 values are required to define a quaternion versus 9 values for rotation matrices, (c) easier to perform smooth interpolations between two quaternions and less prone to numerical drift as it can be normalised to 1 and maintain the rotation constraint (Zhang, 1997). As the pose of the target object remains unchanged, it is regarded as a static Bayesian network. This assumption is realistic considering light contacts and guarded motions during tactile exploration. During exploration of the workspace by performing actions a_t, tactile measurements $\mathbf{z}_t$ are obtained. These measurements are then used to update the current belief of the state. Using a Bayesian formulation:

$$p(\mathbf{x}|\mathbf{z}_{1:t}, \mathbf{a}_{1:t}) = \eta\, p(\mathbf{x}, \mathbf{z}_{1:t}, \mathbf{a}_{1:t}) \tag{4.4}$$

$$= \eta\, p(\mathbf{z}_t|\mathbf{x}, \mathbf{z}_{1:t-1}, \mathbf{a}_{1:t}) p(\mathbf{x}, \mathbf{z}_{1:t-1}, \mathbf{a}_{1:t}) \tag{4.5}$$

where η is a normalization constant. Since $\mathbf{z}_t$ only depends on the action at timestep t and the state, the Eq. (4.5) can be simplified to

$$p(\mathbf{x}|\mathbf{z}_{1:t}, \mathbf{a}_{1:t}) = \eta\, p(\mathbf{z}_t|\mathbf{x}, \mathbf{a}_t) p(\mathbf{x}, \mathbf{z}_{1:t-1}, \mathbf{a}_{1:t}) \tag{4.6}$$

$$= \eta\, p(\mathbf{z}_t|\mathbf{x}, \mathbf{a}_t) p(\mathbf{x}|\mathbf{z}_{1:t-1}, \mathbf{a}_{1:t-1}) \tag{4.7}$$

Note that $p(\mathbf{x}|\mathbf{z}_{1:t-1},\mathbf{a}_{1:t}) = p(\mathbf{x}|\mathbf{z}_{1:t-1},\mathbf{a}_{1:t-1})$, since the state depending on future actions were not considered. The dependence of $\mathbf{x}$ on the actions is solemnly stemming from the measurement model $p(\mathbf{z}_t|\mathbf{x},\mathbf{a}_t)$.

Quaternions are chosen as a smooth representation for the state $\mathbf{x}$ and estimated using a Kalman Filter. To leverage the insights from Eq. (4.3), a linear measurement model is formulated. The Eq.(4.3) can be reformulated using Quaternion algebra as:

$$\widetilde{\mathbf{s}}_{ji} = \mathbf{x} \odot \widetilde{\mathbf{o}}_{ji} \odot \mathbf{x}^* \tag{4.8}$$

where $\odot$ is the Quaternion product, $\mathbf{x}^*$ is the conjugate of $\mathbf{x}$, $\widetilde{\mathbf{s}}_{ji} = \{0, \mathbf{s}_{ji}\}$ and $\widetilde{\mathbf{o}}_{ji} = \{0, \mathbf{o}_{ji}\}$. Since $\mathbf{x}$ is a unit Quaternion, $\sqrt{\mathbf{x} \odot \mathbf{x}^*} = ||\mathbf{x}|| = 1$ can be used to get

$$\widetilde{\mathbf{s}}_{ji} \odot \mathbf{x} = \mathbf{x} \odot \widetilde{\mathbf{o}}_{ji} \tag{4.9}$$

$$\widetilde{\mathbf{s}}_{ji} \odot \mathbf{x} - \mathbf{x} \odot \widetilde{\mathbf{o}}_{ji} = 0 \tag{4.10}$$

Eq. (4.10) can be rewritten using the matrix notation of Quaternion multiplication as:

$$\begin{bmatrix} 0 & -\mathbf{s}ji^T \\ \mathbf{s}_{ji} & \mathbf{s}_{ji}^{\times} \end{bmatrix} \mathbf{x} - \begin{bmatrix} 0 & -\mathbf{o}ji^T \\ \mathbf{o}_{ji} & -\mathbf{o}_{ji}^{\times} \end{bmatrix} \mathbf{x} = \mathbf{0} \tag{4.11}$$

$$\begin{bmatrix} 0 & -(\mathbf{s}_{ji} - \mathbf{o}_{ij})^T \\ (\mathbf{s}_{ji} - \mathbf{o}_{ji}) & (\mathbf{s}_j + \mathbf{s}_i + \mathbf{o}_j + \mathbf{o}_i)^{\times} \end{bmatrix}_{4\times 4} \mathbf{x} = \mathbf{0} \tag{4.12}$$

where $[v]^{\times}$ is the skew-symmetric matrix formed from the vector v. Eq. (4.12) is of the form $\mathbf{H}\mathbf{x} = 0$ where $\mathbf{H}$ is the *pseudo-measurement* matrix such that

$$\mathbf{H} = \begin{bmatrix} 0 & -(\mathbf{s}_{ji} - \mathbf{o}_{ji})^T \\ (\mathbf{s}_{ji} - \mathbf{o}_{ji}) & (\mathbf{s}_j + \mathbf{s}_i + \mathbf{o}_j + \mathbf{o}_i)^{\times} \end{bmatrix} \in \mathbb{R}^{4\times 4} \tag{4.13}$$

The pseudo-measurement matrix $\mathbf{H}$ depends only on the measurements points $\mathbf{s}_{ji}$ and the model points $\mathbf{o}_{ji}$. Furthermore, it can be inferred that, in the no noise case, the true state $\mathbf{x}$ must lie in the nullspace of $\mathbf{H}$.

Eq. (4.12) can be reformulated as a pseudo-measurement model as:

$$\mathbf{H}\mathbf{x} = \mathbf{z}^h \tag{4.14}$$

and enforcing the pseudo-measurements $\mathbf{z}^h = 0$. For each time-step t, $\mathbf{H}_t$ is calculated based on newly obtained measurement points $(\mathbf{s_{ji}})_t$ and transformed model points $(\mathbf{o_{ji}})_t$. If $\mathbf{z}_t$ represents the measurements vector, and $\mathbf{v}_t$ represents the measurement noise at time t,

then:

$$\mathbf{z}_t = \mathbf{z}_t^h + \mathbf{v}_t \tag{4.15}$$

Using Eq. (4.14), the Eq. (4.15) can be reformulated as:

$$\mathbf{H}_t \mathbf{x}_t = \mathbf{z}^h + \mathbf{v}_t \tag{4.16}$$

$$\mathbf{0} = \mathbf{H}_t \mathbf{x}_t - \mathbf{v}_t \tag{4.17}$$

Eq. (4.17) represents a linear equation in the state $\mathbf{x}_t$ with the state-dependent noise term $\mathbf{v}_t$. The measurement noise $\mathbf{v}_t$ is considered as zero mean noise with covariance $\mathbf{P}_t^v$. The exact expressions for the state-dependent measurement noise is defined as follows. Let $\mathbf{P}_t^x$ represent the covariance matrix of state $\mathbf{x}_t$ and $\mathbf{P}_t^v$ the covariance matrix of the measurement noise term $\mathbf{v}$. The equation for the covariance of the measurement noise is similar to (Choukroun et al., 2006) which is motivated from a fundamental proposition from Stochastic Filtering Theory (Jazwinski, 1970)[Chap 3, Pg. 90] as:

$$\mathbf{P}_t^v = \frac{1}{4}\rho \left[tr(\hat{\mathbf{x}}_{t-1}\hat{\mathbf{x}}_{t-1}^T + \mathbf{P}_{t-1}^x)\mathbb{I}_4 - (\hat{\mathbf{x}}_{t-1}\hat{\mathbf{x}}_{t-1}^T + \mathbf{P}_{t-1}^x) \right] \tag{4.18}$$

where ρ is a constant which corresponds to the uncertainty of the correspondence measurements, tr refers to trace and $\hat{\mathbf{x}}_t$ refers to the mean of $\mathbf{x}_t$ at time t. The process model is given as $\mathbf{x}_t = \mathbf{x}_{t-1}$ as it represents the time-invariant rotation estimate of the object.

Hence, the Kalman filter equations can be defined as follows:

$$\mathbf{x}_t = \mathbf{x}_{t-1} - \mathbf{K}_t \left(\mathbf{H}_t \mathbf{x}_{t-1} \right) \tag{4.19}$$

$$\mathbf{P}_t^x = \left(\mathbf{I} - \mathbf{K}_t \mathbf{H}_t \right) \mathbf{P}_{t-1}^x \tag{4.20}$$

$$\mathbf{K}_t = \mathbf{P}_{t-1}^x \mathbf{H}_t^T \left(\mathbf{H}_t \mathbf{P}_{t-1}^x \mathbf{H}_t^T + \mathbf{P}_t^v \right)^{-1} \tag{4.21}$$

where $\mathbf{x}_{t-1}$ is the state estimate at $t-1$, $\mathbf{K}_t$ is the Kalman gain and $\mathbf{P}_{t-1}^x$ is the covariance matrix of the state at $t-1$.

Since the Kalman filter does not preserve the constraints on the state-variables such as the unit-norm property of the quaternion, a common technique is to normalise the state and the associated covariance matrix after each update as:

$$\mathbf{x}_t \leftarrow \frac{\mathbf{x}_t}{||\mathbf{x}_t||_2} \quad \mathbf{P}_t^x \leftarrow \frac{\mathbf{P}_t^x}{||\mathbf{x}_t||_2^2} \tag{4.22}$$

The rotation estimate $\mathbf{x}$ (quaternion) can be converted to the rotation matrix form $\mathbf{R} \in SO(3)$ and inserted into Eq. (4.1). Once the rotation estimate $\mathbf{R}$ is found, the translation estimate

at time t $\mathbf{t}_t$ is computed in closed form:

$$\mathbf{t}_t = \frac{1}{N}\sum_{i=0}^{N}(\mathbf{s}_i - \mathbf{R}_t\mathbf{o}_i) \tag{4.23}$$

Thus, with each iteration of the TIQF, a new rotation and translation estimate is obtained that is used to transform the model. The transformed model is used to recompute correspondences and repeat the TIQF update steps. The change in homogeneous transformation between iterations is calculated for convergence check i.e., $\Delta_{TIQF} < \xi^{conv}$ if the difference in the output pose is less than a specified threshold which in the experiments is 0.1 mm and 0.1° respectively and/or maximum number of iterations in order to check for convergence ($max\_it_{TIQF} = 100$).

4.2.4 Active Touch Exploration

To reduce the number of touches required to converge to the true position of the object, informed decisions on which actions $\mathbf{a}_t$ to perform next based on the current state estimate were necessary. The set of possible actions was constrained by the position of the object in the workspace and the reachability of that position by the robot, and was denoted as $\mathcal{A}$. The set of actions $\mathcal{A}$ were generated by sampling uniformly along the faces of a bounding box on the current estimate of the pose. An action is defined as a ray represented by a tuple $\mathbf{a} = (\mathbf{n}, \mathbf{d})$, with $\mathbf{n}$ as the start point and $\mathbf{d}$ the direction of the ray. The objective was to choose the action $\mathbf{a}_t^*$, that *maximizes* the overall expected *Information Gain*. The information gain was measured as the Kullback–Leibler (KL) divergence between the posterior distribution $p(\mathbf{x}|\mathbf{z}_{1:t}, \mathbf{a}_{1:t})$ after executing action $\mathbf{a}_t$ and the prior distribution $p(\mathbf{x}|\mathbf{z}_{1:t-1}, \mathbf{a}_{1:t-1})$. However, it should be noted that future measurements denoted as $\hat{\mathbf{z}}_t$ were hypothetical as the robot action has not yet been performed at time t. The action-measurement model $p(\hat{\mathbf{z}}_t|\mathbf{x}, \mathbf{a}_t)$ was approximated as a ray-mesh intersection in simulation to extract the hypothetical measurement given a certain action when the object is at the estimated pose. For each hypothetical action $\hat{\mathbf{a}}_t \in \mathcal{A}(\mathbf{x}_{t-1})$ and the hypothetical measurement $\hat{\mathbf{z}}_t$, the posterior is estimated by $p(\mathbf{x}|\hat{\mathbf{z}}_{1:t}, \hat{\mathbf{a}}_{1:t})$ known as the *one-step look ahead*. Therefore, the most optimal action $\mathbf{a}_t^*$ was given by

$$\mathbf{a}_t^* = \arg\max_{\hat{\mathbf{a}}_t} \int_{\mathbf{x}} p(\mathbf{x}|\hat{\mathbf{z}}_{1:t}, \hat{\mathbf{a}}_{1:t}) \log \frac{p(\mathbf{x}|\hat{\mathbf{z}}_{1:t}, \hat{\mathbf{a}}_{1:t})}{p(\mathbf{x}|\mathbf{z}_{1:t-1}, \mathbf{a}_{1:t-1})} d\mathbf{x} \tag{4.24}$$

Algorithm 1: Active touch for tactile point cloud registration and accurate object localization

Input: ${}^{v}\mathbf{x}$, ${}^{v}\Sigma^{x}$, ${}^{v}\mathbf{t}$, $\mathcal{O}$
Result: ${}^{t}\mathbf{x}$, ${}^{t}\Sigma^{x}$, ${}^{t}\mathbf{t}$
Initialisation:
$\mathbf{x}_t \leftarrow {}^{v}\mathbf{x}$, $\Sigma^{x}_t \leftarrow {}^{v}\Sigma^{x}$, $\mathbf{t}_t \leftarrow {}^{v}\mathbf{t}$
Measurements ${}^{t}\mathcal{S} \leftarrow \{\}$, Correspondences $\mathcal{C} \leftarrow \{\}$,
Actions $\mathcal{A} \leftarrow \{\}$, Sim. Measurements $\mathcal{Z} \leftarrow \{\}$,
KL Divergence $\mathcal{D}_{KL} \leftarrow \{\}$;
while $\Delta_{TIQF} > \xi^{conv}$ *or it* $\neq$ *100* **do**
  $\hat{\mathcal{O}} \leftarrow$ transform($\mathcal{O}$, $\mathbf{x}_t$, $\mathbf{t}_t$) ;
  if *size*(${}^{t}\mathcal{S}$)≤ 2 **then**
    $\mathbf{a}^{*}_t$ = select_random_action($\hat{\mathcal{O}}$) ;
  else
    $\mathcal{A} \leftarrow$ generate_possible_actions($\hat{\mathcal{O}}$) ;
    $\mathcal{Z} \leftarrow$ simulate_measurements($\mathcal{A}, \hat{\mathcal{O}}$) ;
    $\mathcal{D}_{KL} \leftarrow \{\}$;
    for $\hat{\mathbf{z}}_t$ *in* $\mathcal{Z}$ **do**
      ${}^{t}\hat{\mathcal{S}} \leftarrow {}^{t}\mathcal{S} \cup \{\hat{\mathbf{z}}_t\}$;
      $\hat{\mathcal{C}}$ = estimate_correspondences($\hat{\mathcal{O}}$, ${}^{t}\hat{\mathcal{S}}$) ;
      $\hat{\mathbf{x}}_t, \hat{\Sigma}^{\mathbf{x}}_t \leftarrow$ update_TIQF($\mathbf{x}_t$, $\Sigma^{\mathbf{x}}_t$, $\hat{\mathcal{C}}$) ;
      $KL \leftarrow$ compute_kl_div($\mathbf{x}_t$, $\Sigma^{\mathbf{x}}_t$, $\hat{\mathbf{x}}_t$, $\hat{\Sigma}^{\mathbf{x}}_t$) ;
      $\mathcal{D}_{KL} \leftarrow \mathcal{D}_{KL} \cup KL$;
    end
    $\mathbf{a}^{*}_t \leftarrow$ choose_best_action($\mathcal{A}$, $\mathcal{D}_{KL}$) ;
    $\mathbf{z}_t \leftarrow$ execute_action($\mathbf{a}^{*}_t$) ;
    ${}^{t}\mathcal{S} \leftarrow {}^{t}\mathcal{S} \cup \{\mathbf{z}_t\}$;
    $\mathcal{C}$ = estimate_correspondences($\hat{\mathcal{O}}$, ${}^{t}\mathcal{S}$) ;
    $\mathbf{x}_t, \Sigma^{x}_t \leftarrow$ update_TIQF($\mathbf{x}_t$, $\Sigma^{\mathbf{x}}_t$, $\mathcal{C}$) ;
    $\mathbf{t}_t \leftarrow$ compute_translation($\mathbf{x}_t$, $\mathcal{C}$) ;
  end
  $it++$;
end

Given that the prior and posterior are multivariate Gaussian distributions, the KL divergence in Eq. (4.24) can be computed in closed form as (derivation provided in Appendix B):

$$\mathbf{a}^{*}_t = \arg\max_{\hat{a}_t} \frac{1}{2}\Bigg[log\frac{det(\bar{\Sigma}_{t-1})}{det(\hat{\bar{\Sigma}}_t)} + Tr(\bar{\Sigma}^{-1}_{t-1}\hat{\bar{\Sigma}}_t)) - d + (\hat{\bar{\mathbf{x}}}_t - \bar{\mathbf{x}}_{t-1})^{T}\bar{\Sigma}^{-1}_t(\hat{\bar{\mathbf{x}}}_t - \bar{\mathbf{x}}_{t-1})\Bigg] \tag{4.25}$$

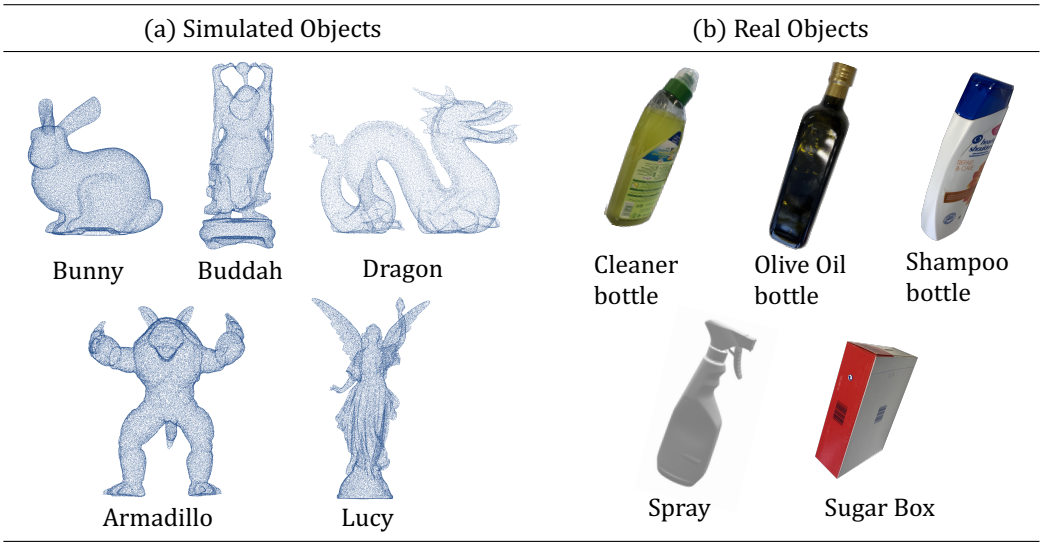


Figure 4.3: (a) Simulated objects from Stanford 3D Scanning Repository (Levoy et al., 2005) and (b) real experimental objects

where d is the dimension of the state vector and $d = 4$ in this case. This enables to evaluate an exhaustive list of actions at marginal computation cost in *real time* without the need to prune actions or setting trade-offs with computation time as compared to prior work (Saund et al., 2017). Once the optimal action was calculated using Eq. 4.24 in simulation, the selected action was performed by the robot and real measurement $\mathbf{z}_t$ was extracted. The overall algorithm is shown in Algorithm. 1.

4.3 Experimental Results

4.3.1 Experimental Setup

The experimental setup shown in Fig. 4.1 consists of Universal Robots UR5 robot with a Robotiq 2F140 Gripper. The tactile sensors from XELA Robotics (Tomo, 2019) were used on the fingertips and the phalanges as described in Chap. 3. A contact was established with the object when the norm of the 3-axis force value of any taxel f_r exceeds a threshold τ_f, which is defined with respect to the baseline values and had been tuned empirically. The 3D positions of the contacted taxels are transformed into the robot base frame using robot kinematics and are appended to the tactile point cloud ${}^{t}\mathcal{S}$. An Azure Kinect DK RGB-D camera was placed in front of the workspace, which provided the vision point cloud ${}^{v}\mathcal{S}$. The simulation and experimental results are provided in the following sections. All simulation and real experiments were executed on a workstation running Ubuntu 18.04 with 8 core Intel i7-8550U CPU @ 1.80GHz and 16 GB RAM.

4.3.2 Simulation Results

In order to validate the proposed method, extensive simulation experiments were performed using the Stanford 3D Scanning Repository (Levoy et al., 2005). Five different CAD objects from (Levoy et al., 2005) was used namely: Bunny, Buddah, Dragon, Armadillo and Lucy as shown in Fig. 4.3a. Correspondences between the model and scene point clouds were unknown *a priori* similar to realistic scenarios. All the models were scaled to within $[0,1]^3$ cm box. Noise randomly sampled from a normal distribution $\mathcal{N}(0, 5 \times 10^{-3})$ was added to the cloud obtained from the meshes, henceforth called *scene* point cloud. The initial pose for each model was sampled uniformly from [-5cm, 5cm] and $[-30^\circ, 30^\circ]$ for position and orientation, respectively. The initial state $\mathbf{x}_0$ was obtained from the initial start pose and the initial covariance $\Sigma_0^{\mathbf{x}}$ was set to $\mathbb{I}_4$. To simulate tactile measurements, points were sequentially sampled from the *scene* point cloud and registered to the model cloud using the proposed TIQF algorithm. Random sampling was compared with active sampling of points. Each experiment was repeated 100 times for each of the five objects. Actions were uniformly sampled on each face of the bounding box encapsulating the scene and ray-mesh (triangle) intersection algorithm was used in order to extract the measured points. The Möller–Trumbore intersection algorithm (Möller and Trumbore, 1997) was used in order to perform the ray-mesh intersection to extract hypothetical measurements. For random action selection, an action was randomnly selected from the sampled set of actions and was executed. For active touch selection, hypothetical measurements $\hat{\mathbf{z}}$ were extracted using the generated actions and one-step lookahead for each action-hypothetical measurement pair was performed by running the TIQF algorithm for a fixed number of iterations. The optimal action $\mathbf{a}_t^*$ was chosen which was associated with the largest KL divergence of the hypothetical posterior with the prior belief. For all the objects in simulation, a total of 100 possible actions were generated in order to choose the optimal action at each measurement step. Furthermore, it was noted that due to the low number of sparse points available for registration, the TIQF algorithm often gets into local minima. To address this, a well-known strategy was used to add local perturbations sampled from a uniform distribution $[-2^\circ, 2^\circ]$ around the local minima. The action sampling is demonstrated in Fig. 4.4a and the registration output is shown in Fig. 4.4c for the Bunny dataset from the Stanford Scanning Repository. The simulation results are reported in Fig. 4.5 showing the root mean square error (RMSE) of translation and rotation versus number of points.

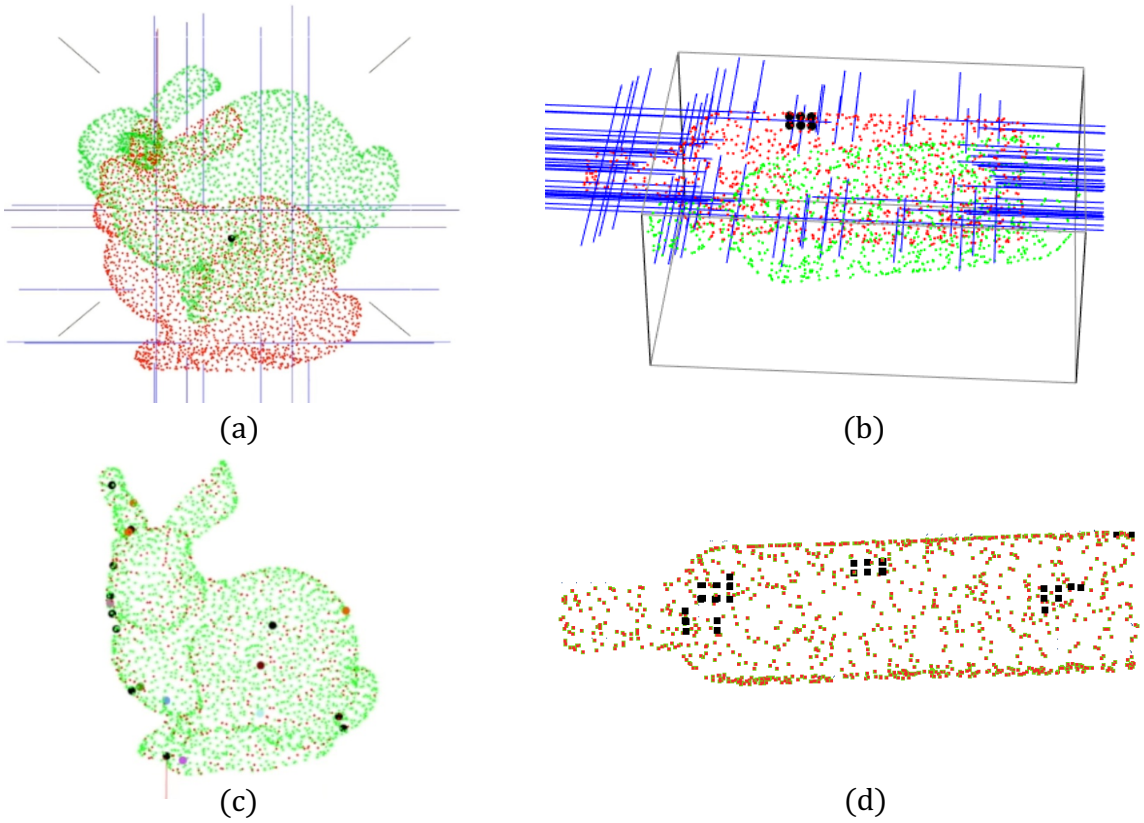


Figure 4.4: Action sampling (a), (b) and point cloud registration output from TIQF (c) and (d). The initial point cloud are shown in green and target point cloud in red. The black points are chosen through action sampling and used to register with the target point cloud through sparse-to-dense point cloud registration with TIQF.

4.3.3 Robot Experimental Results

In order to validate the framework with robotic systems, five daily objects of various intrinsic properties were chosen as shown in Fig. 4.3b. The following objects were used: shampoo, sugar box, spray, cleaner, and olive oil bottle. The objects were chosen according to the following criteria: varying shape between simple (cuboid, cylinder) to complex (for instance, spray) and varying degrees of transparency (for instance, highly transparent cleaner, highly opaque sugar box). The corresponding CAD meshes for the real objects were obtained using a high-precision 3D scanner. The objects are rigidly attached to the workspace and the ground truth was extracted with respect to the world frame $\mathcal{W}$ with a marker-based system. The objects were randomly moved around the workspace between experiments to evaluate the robustness of the approach. The robot actions were performed as guarded motions so that the robots do not topple the other objects in the workspace. The initial estimate was computed from the vision point cloud by using the TIQF algorithm. For

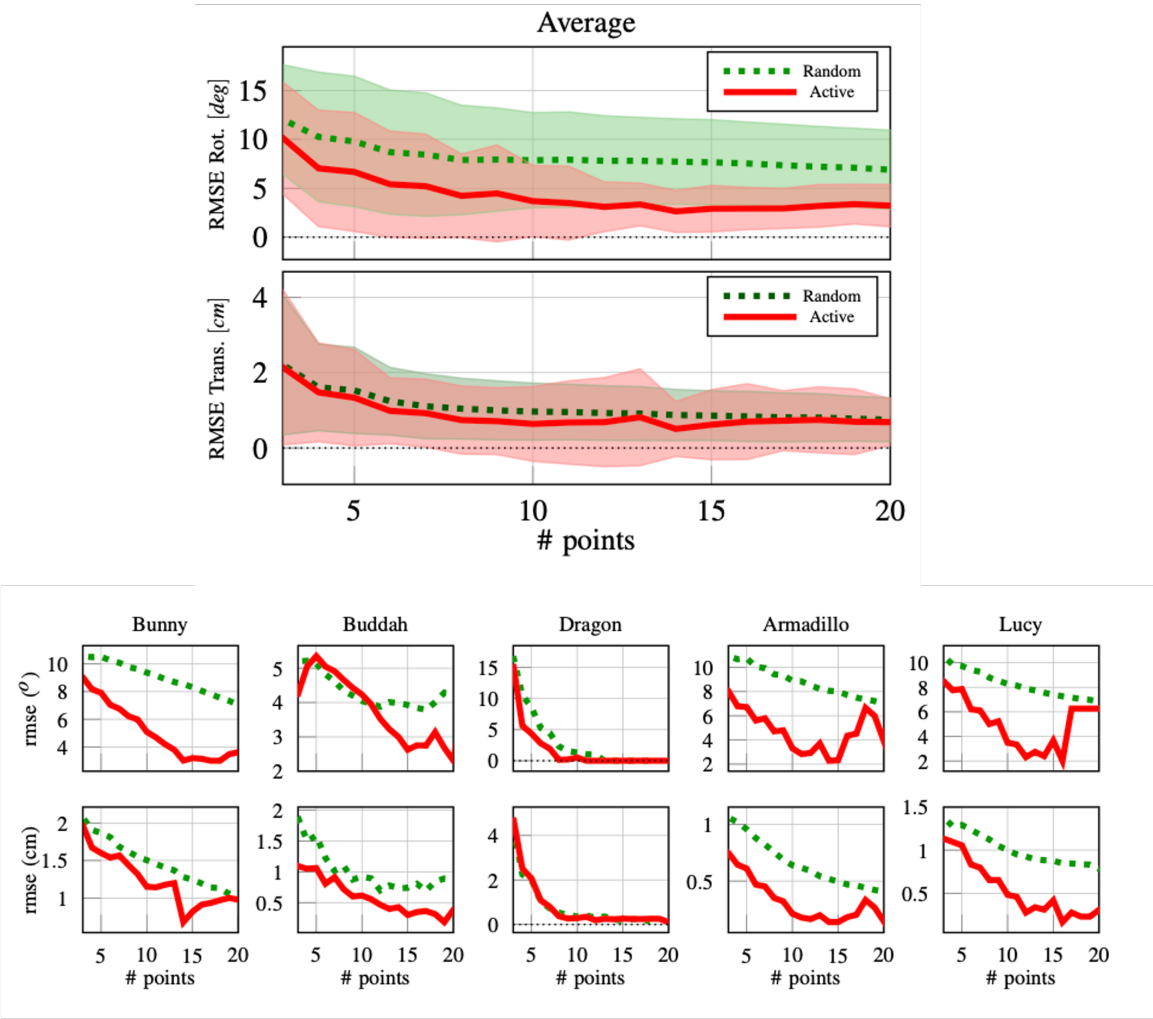


Figure 4.5: Simulation experiments on five meshes from the Stanford Scanning Repository

the tactile-based pose refinement, the tactile actions were generated uniformly with directions along coordinate axes on the 5 faces of the bounding box around the current estimate, assuming that it is unfeasible to contact the object from the bottom when placed on a table. The action list was pruned in order to remove actions that were kinematically unfeasible and that collide with the workspace. Hypothetical actions-measurement pairs were generated with the ray-mesh intersection with the current estimate of the object. The candidate action with the highest expected information gain with one-step lookahead was chosen and performed on the real object. As the actions may not contact the real objects as they are based on the current estimate, it must be noted that *negative information* i.e., information about absence of measurements was not considered in the active perception calculation. However, since the action generation and selection was guided initially by the vision estimate and iteratively updated with the tactile measurements, empirically it was found that

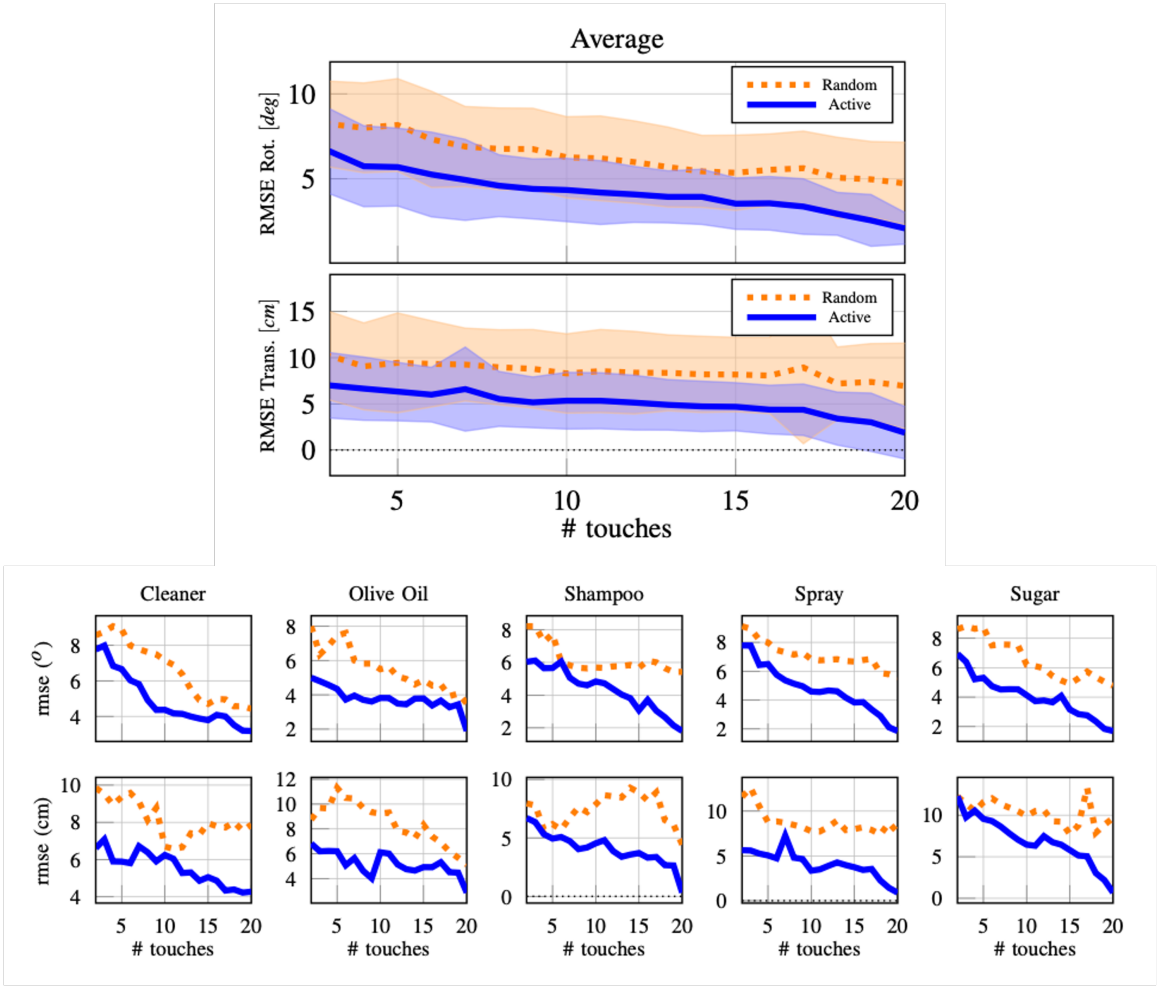


Figure 4.6: Robot experiments on five selected daily objects

fewer actions resulted in negative information. The action sampling for the olive oil bottle is shown in Fig. 4.4b and consequently, the final pose estimate with TIQF is shown in Fig. 4.4d. The ROS MoveIt motion planner was used which ensured that the robot moves safely to start positions of actions by moving over the workspace W_{XYZ} at a height larger than the biggest object and descends vertically to the start point of the selected action. The experimental trials for the five objects was repeated ten times for each object by varying their ground truth locations. The results of the experiments are presented in Fig. 4.6.

4.4 Discussion

In this chapter, a novel method, termed the Translation-Invariant Quaternion Filter (TIQF) for object pose estimation has been presented. For both simulated objects and real objects, a highly accurate pose estimate (< 2 cm and $< 2^{\circ}$ RMSE error) was achieved. As seen in Fig. 4.5, around 20 points from the *scene* point clouds were sufficient to con-

verge to the target pose within the aforementioned error. It should be noted that the *model* point clouds sampled from the CAD model contained approximately 2000 points. This highlights the benefit of the TIQF algorithm for sparse-to-dense point cloud registration. Similarly, for real objects, the pose error converged within 20 tactile probing actions. The number of points acquired from each touch can vary between 0-24 points due to the varying possibility of the taxels contacting the object. The variation in shape complexity did not affect the performance of TIQF algorithm as seen from Fig. 4.6. Taking into account the performance of active perception, Fig. 4.5 shows that in all simulated objects, the proposed active strategy outperformed the random strategy (used in state-of-the-art methods such as (Vezzani et al., 2017)) in terms of accuracy (average RMSE for rotation and translation) and convergence rate with respect to the number of points for both rotation and translation. Furthermore, the experimental results show that in the first 5 measurements, the RMSE for translation and rotation for the active strategy was markedly lower than that of the random approach. This demonstrated that the proposed method performs effectively right from the first touch. The results in simulation were corroborated with the experiments with the selected daily objects as seen in Fig. 4.6. Moreover, the objects were intentionally chosen with varying degree of transparency, such as the cleaner that caused issues for vision sensor but was accurately localised with tactile sensing. As noted earlier, the proposed method can reason over multiple candidate actions to find the most optimal action using very low computation time without the need for high compute hardware. This is shown in Tab. 4.1 for meshes with 1000 and 5000 triangular faces respectively wherein the one-step lookahead time was calculated. It was found that for an object mesh with 1000 triangular faces, less than 2s was needed to compute for 100 actions. In comparison, state-of-the-art works such as (Tosi et al., 2014) constrain the action selection search policy with computation time, as their approach would consume far too long ($\sim 10s$ for a simple cuboid object) for practical robotic applications.

To summarise, the robotic system using the proposed active touch-based approach

Table 4.1: The computation time required for action generation and action selection with the one-step look ahead for mesh with 5000 *triangular faces and mesh with* 1000 *triangular faces. The performance shown here is representative as it is dependent on chosen hardware. The values are deterministic for chosen hardware and mesh size.*

# Actions	**5000 triangular faces (s)**	**1000 triangular faces (s)**
10	0.33	0.17
100	5.06	1.75
1000	42.56	13.70

guided by a vision estimate, accurately and efficiently estimated the pose of objects in an unknown workspace. Moreover, the vision estimate was corrected by using the tactile modality. The proposed method enables the robotic system to *actively* reason upon possible next actions and choose the next best touch based on an information gain metric. It is demonstrated that using the active touch point selection, on average highly accurate results can be achieved with fewer measurements. TIQF was capable of handling sparse and sequential data such as tactile data and dense data provided in a batch manner such as vision data effectively for pose estimation.

There are certain limitations in this chapter: the point cloud representing the object (typically extracted from a CAD) mesh was needed for performing registration. TIQF was susceptible to get stuck in local minima if incorrectly initialised. These drawbacks are overcome with the Stochastic-TIQF (S-TIQF) algorithm presented in Chap. 5. Moreover, TIQF and S-TIQF are rigorously compared against state-of-the-art baselines in various benchmark datasets and real robotic experiments in Chap. 5. Furthermore, time-invariant pose estimation is focussed in this chapter where the objects did not move during robot interaction. A time-dynamic visuo-tactile pose tracking algorithm named ArtReg is presented in Chap. 6.

Chapter 5

S-TIQF: Shared Visuo-Tactile Interactive Perception for Robust Pose Estimation

Parts of this chapter are **published** as:

- "*Shared Visuo-Tactile Interactive Perception for Robust Object Pose Estimation*", **P. K. Murali**, B. Porr, and M. Kaboli, in The International Journal of Robotics Research (IJRR) (2024): 02783649241301443 (Murali et al., 2024).
- "*Active Visuo-Tactile Interactive Robotic Perception for Accurate Object Pose Estimation in Dense Clutter*," **P. K. Murali**, A. Dutta, M. Gentner, E. Burdet, R. Dahiya, and M. Kaboli in IEEE Robotics and Automation Letters (RA-L), vol. 7, no. 2, pp. 4686-4693, April 2022 (Murali et al., 2022b).

The video of the experiments from this chapter is available here: `https://drive.google.com/file/d/1h4mUDNETPOTMIFYyzASjbsTpb-ClNj5Q/view?usp=sharing`video-link

5.1 Introduction

Humans are capable of seamlessly integrating perceptual information from vision and touch (haptic) to maintain a high-level of cognitive understanding of the environment (Hatwell, 1987; Ernst and Banks, 2002). Robots should also be able to achieve a similar level of scene understanding given that they are similarly equipped, for example, with visual and tactile sensing. The shared perception among complementary sensing modalities

offers a comprehensive and accurate scene representation as well as addressing the weaknesses inherent in individual sensor systems. This should make perception robust against sensor failure, as the robot could then rely on the other modality to retain the same level of functionality (Murali et al., 2022e). As with humans, robots have also the option to enhance their perceptual information through purposeful manipulative actions, a technique known as interactive perception which forges a symbiotic relationship between action and perception (Bohg et al., 2017). Thus, by leveraging shared perception, robots can potentially improve their perceptual scene understanding which can improve the effectiveness of downstream applications such as autonomous manipulation and interaction in unknown environments.

However, sharing multi-modal visuo-tactile perceptual information is challenging due to the weakly paired and complementary nature of the sensing modalities. Visual perception provides dense and global information of the scene, whereas tactile perception provides sparse and local contact information. Temporal misalignment also affects shared perception as visual data can be captured in one shot while tactile data acquisition requires sequential contact interactions with objects (Li et al., 2020a). Previous research in the realm of multi-robot and multi-sensor shared perception frequently relies on employing identical sensing modalities, typically using multiple cameras. This simplification streamlines the representation of the shared scene (Lauri et al., 2020). Active perception techniques, characterized by the proactive selection of sensor positions to enhance information gathering, are often utilized in the context of single sensor setups or setups with multiple sensors of the same modality (Connolly, 1985; Delmerico et al., 2018). Nevertheless, the extension of active perception methods to multi-sensor configurations comprising different modalities, such as visual and tactile sensing, poses a non-trivial challenge. Similarly, there are recent works tackling the problem of category-level object pose estimation wherein the exact CAD models of object of interest are unknown but prior knowledge of objects belonging to the same category is available. These works typically regress a shared canonical representation of all possible object instances within a category and use the measured depth information to lift from 2D to 3D space to perform object pose estimation (Wang et al., 2019; Deng et al., 2022; Lee et al., 2021). In summary, the state-of-the-art methods have several limitations: (a) category-level pose estimation techniques are predominantly tailored to visual sensing information (RGB and depth data), rendering them unsuitable for direct adaptation to other sensory modalities, such as tactile sensing; (b) these methods are also not evaluated for photometrically challenging objects such as transparent objects (Wang et al., 2022a); (c) active perception methods that are designed for mono-modal settings cannot be directly extended to multi-modal settings; (d) misalignment between multi-modal visual and tactile

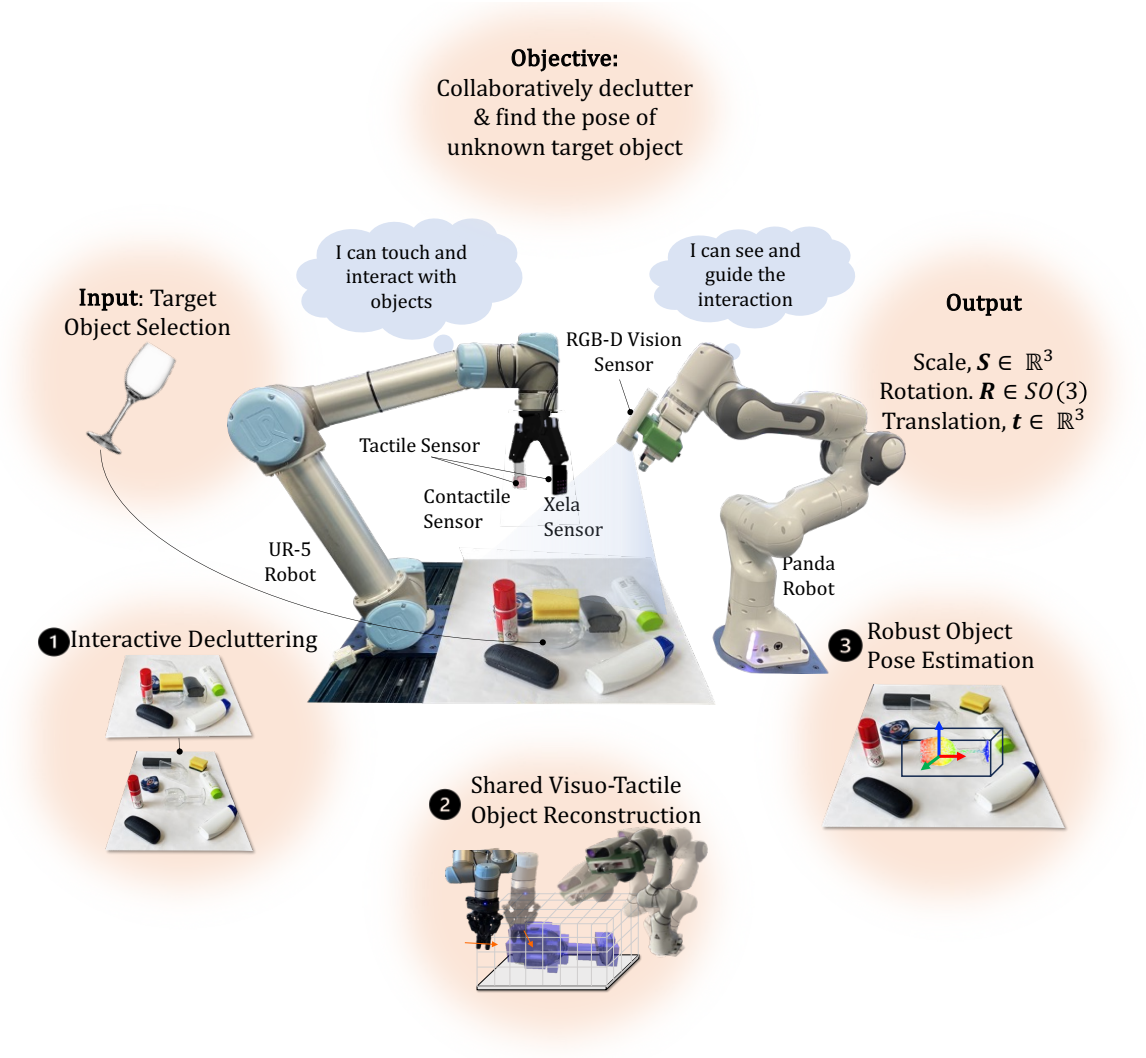


Figure 5.1: Experimental setup: A Universal Robots UR5 sensorised with tactile sensor arrays on the Robotiq Gripper, a Franka Emika Panda robot equipped with a Azure Kinect RGB-D camera and clutter objects containing the novel target object. The objective is to collaboratively declutter the scene, share the visuo-tactile perceptual information and find the target pose of the object.

data often arises due to calibration errors that affects the shared perceptual information. Addressing the misalignment requires specific calibration procedures, which are often laborious and time consuming.

In Chap. 4, a recursive Bayesian filtering approach for object pose estimation through point cloud registration termed the *translation-invariant Quaternion filter (TIQF)* was presented. However, TIQF assumes a priori knowledge of the CAD model of the target object and is prone to get stuck in local minima if incorrectly initialized. In this chapter, the limitations of TIQF have been overcome and several new contributions are presented as

follows:

I A novel shared visuo-tactile perception method for scene representation and object reconstruction through a data-efficient joint information-theoretic approach for active perception (vision or tactile).

II An improved approach termed *Stochastic Translation-Invariant Quaternion Filter (S-TIQF)* which is a recursive Bayesian filtering method with robust stochastic optimization for global optimal pose estimation. S-TIQF estimates the 6 DoF pose and 3 DoF scale of unknown instances of categorical objects and relaxes the need for prior known model of the object.

III A necessary condition for shared perception is the accurate calibration between the sensing modalities. A novel approach for *in-situ* visuo-tactile based hand-eye calibration is presented using arbitrary objects which removes the constraint of specific hand-eye calibration targets and time-consuming calibration procedures.

IV The developed methods are integrated into a full-fledged framework that enables multi-robot teams to share their perceptual information with the objective to declutter a complex scene, reconstruct and robustly estimate the pose of objects.

Extensive experiments were conducted that validated the proposed framework against state-of-the-art approaches on different benchmark datasets and real robotic setup (Fig. 5.1).

5.2 Methodology

5.2.1 Problem Formulation and Framework

The objective is to accurately identify the rotation $\mathbf{R} \in SO(3)$, position $\mathbf{t} \in \mathbb{R}^3$ and scale $\mathbf{S} \in \mathbb{R}^3$ of an *unknown* target object in dense clutter by sharing the scene perception between visual and tactile sensing. The target object belongs one of the N_{c_O} known categories of the objects and can be opaque or transparent. However, no knowledge of the object model is assumed *a priori*. Firstly, in order to identify the individual rotations and positions of objects the robots autonomously and deterministically declutter the scene through interactive perception (Fig. 5.2a). Secondly, vision and tactile sensing are used by the robots to extract a shared scene representation and explore the unknown object for reconstruction (Fig. 5.2b). Finally, the reconstructed object model is used for pose estimation using the S-TIQF algorithm with sensor acquired point cloud (Fig. 5.2c). Furthermore, if there

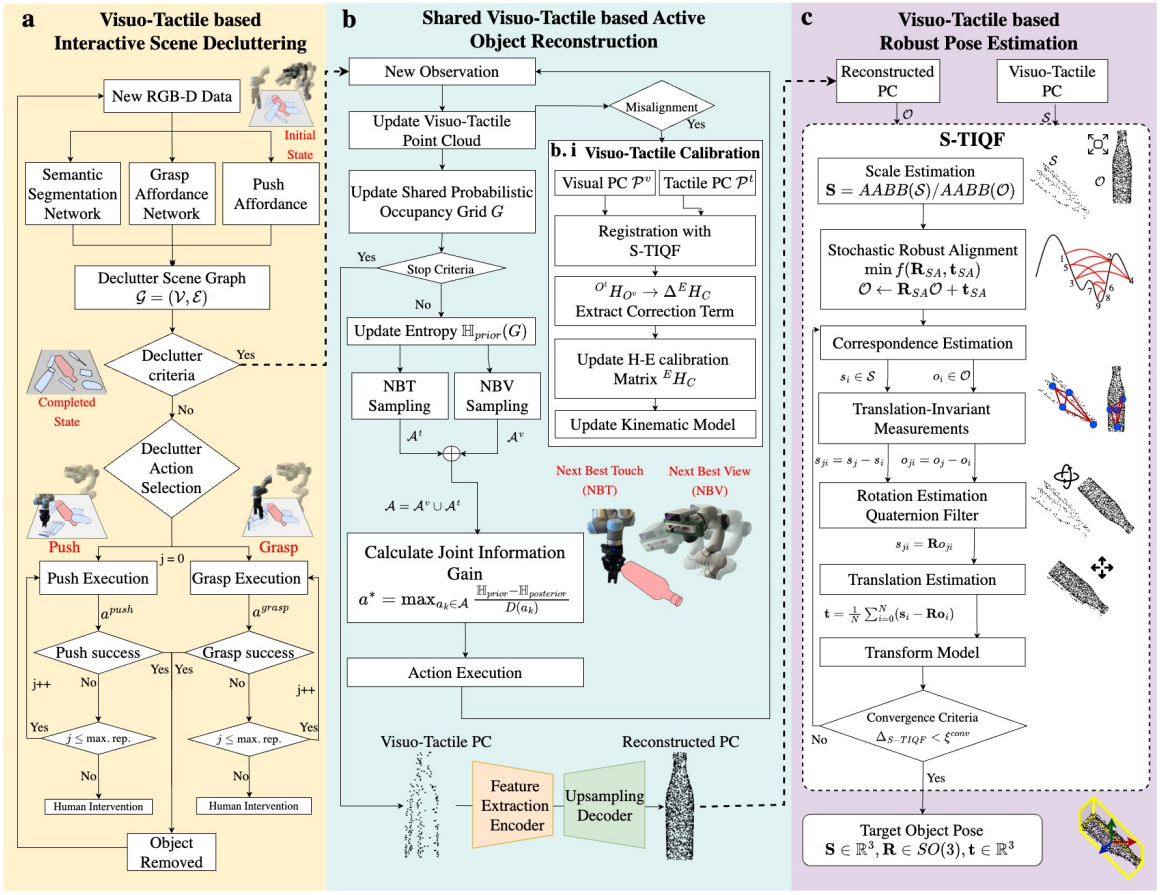


Figure 5.2: The proposed framework for interactive visuo-tactile shared perception for object reconstruction and pose estimation in dense clutter.

is a discrepancy between the visual and tactile point cloud data, it is typically due to incorrect hand-eye calibration and a novel in-situ visuo-tactile hand-eye calibration solution employing again the S-TIQF algorithm is also presented (Fig. 5.2bi).

5.2.2 Visuo-Tactile based Interactive Scene Decluttering

Before identifying the rotation and position of individual objects with the S-TIQF algorithm, it may be necessary to declutter a possibly cluttered scene. As objects may be present in random configurations in the scene, a method and a formalism are necessary to encode the *spatial* and *support relationships* between the objects. Such relationships are encoded in the form of a *scene graph* termed *declutter graph*.

The declutter graph is a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where the vertices of the graph $\mathcal{V}$ represent the objects in the scene and the edges $\mathcal{E}$ from $v_i \in \mathcal{V}$ to $v_j \in \mathcal{V}$ encodes explicitly the actions needed for decluttering or singulating object v_j from object v_i. Implicitly, the edges represent the spatial and support relationships between the objects in $\mathcal{V}$. The root

node of the graph represents the target unknown object O_T which is used to identify the shape and pose. The declutter graph is constructed based on outputs from a semantic segmentation network and grasp affordance network as shown in Fig. 5.2a. An RGB image and a depth image are taken as inputs for the semantic segmentation network and grasp affordance network respectively. Off-the-shelf semantic segmentation network from Chen et al. (2017) and grasp affordance network from Morrison et al. (2020) were used. The pre-trained models were fine-tuned with the real-world object datasets in clutter and their respective segmentation masks. The output of the semantic segmentation network $\mathcal{M}_{seg}$ provides the various objects present in the scene $O_k \in \mathcal{O}$. It is posited that the scene contains a singular instance of the target object, while the clutter objects are characterized by other categories distinct from the target object to facilitate the decluttering process. The edges of the graph are extracted through overlap or proximity metric from $\mathcal{M}_{seg}$ defined as follows.

Definition 1 *Overlap Metric: Two objects representing vertices of the graph v_i, v_j constitute an edge $e_{ij} \in \mathcal{E}$ if the overlap measure is greater than the threshold μ_o. The overlap measure is defined as the Intersection over Union (IoU) value i.e., $IoU_{ij} = (\mathcal{C}_i \cap \mathcal{C}_j)/(\mathcal{C}_i \cup \mathcal{C}_j)$ defines all points in the minimum area bounding box of the respective object masks.*

Definition 2 *Proximity Metric: Two objects representing vertices of the graph v_i, v_j constitute an edge $e_{ij} \in \mathcal{E}$ if the proximity measure is less than the threshold μ_d. The proximity measure d_{ij} is defined as the shortest distance between the two contours of the object masks.*

Thus, an edge $e_{ij} \in \mathcal{E}$ is given by

$$e_{ij} = \begin{cases} IoU_{ij} & (IoU_{ij} > \mu_o) \\ 1/d_{ij} & (d_{ij} < \mu_d) \wedge (IoU_{ij} \leq \mu_o) \\ 0 & \text{otherwise} \end{cases} \tag{5.1}$$

Each edge $e \in \mathcal{E}$ also has an action attribute attached with them. The grasp affordance network provides a grasp action a^{grasp} and the grasp quality measure q_k as output which defines the edge attribute. For an edge e_{ij} directed from v_i to v_j, the edge e_{ij} is attributed with a grasp or a push action for object v_j based on a grasp threshold μ_q as:

$$a_k = \begin{cases} a_k^{grasp} & q_k \geq \mu_q \\ a_k^{push} & q_k < \mu_q \end{cases} \tag{5.2}$$

Hence, the declutter scene graph encodes the *next object to singulate* and the *action (prehensile/ non-prehensile)* to perform. It ensures a targeted and greedy approach to separate

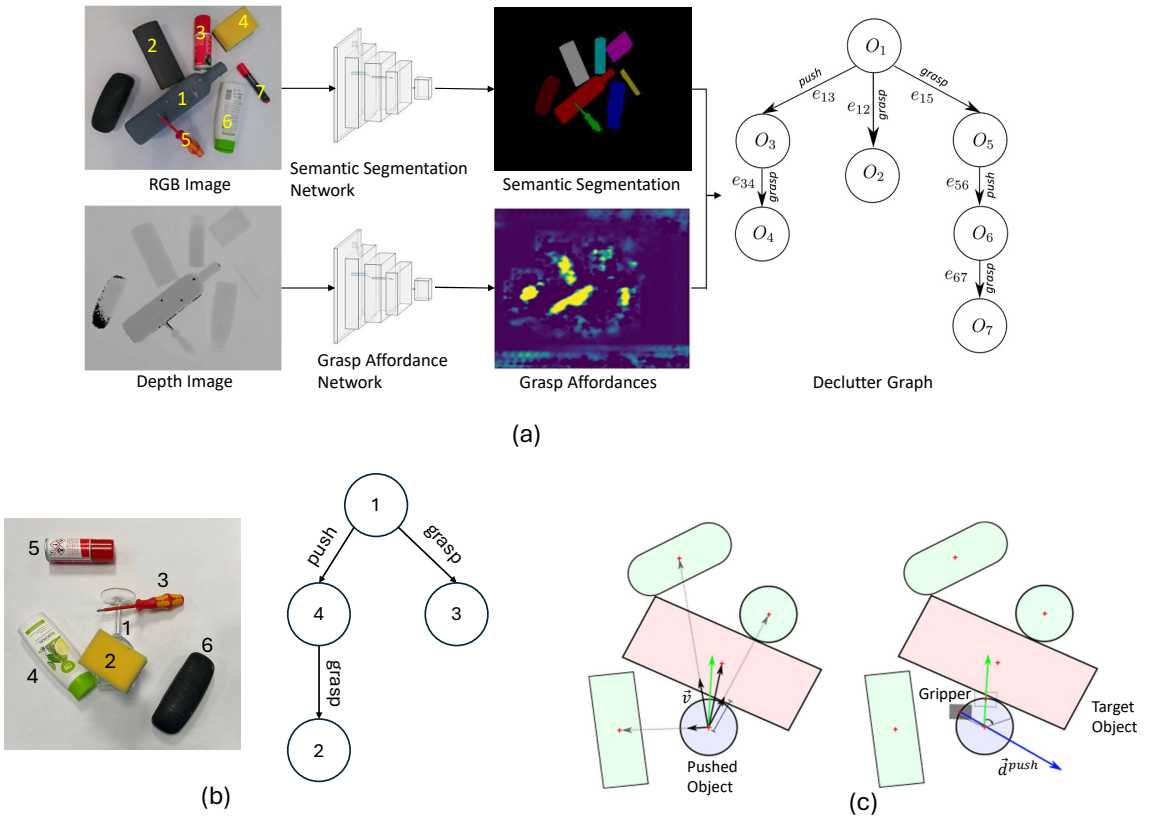


Figure 5.3: (a) Pipeline for the declutter graph from the semantic segmentation network and the grasp affordance network. (b) Another example of declutter graph with a transparent wineglass as target object. (c) Push action formulation.

out all cluttering objects around the target object. The graph is traversed in a depth-first search manner. If there are no children nodes to the root node in the declutter graph $\mathcal{G}$, then the decluttering procedure is complete. Utilising prehensile (grasping) and non-prehensile (push) actions allows the robotic system to choose the action that is more confident and increases the flexibility of the system. Fig. 5.3a,b shows declutter scene graphs for different target objects with the associated grasp/ push action for each object. In the experiments, the threshold values are set as follows: $\mu_o = 0.05, \mu_d = 0.5, \mu_q = 0.1$. As shown in Fig. 5.2a, after each action was performed to singulate an object, the graph was updated and the process was repeated until the decluttering was completed.

Grasp Action for Decluttering

The grasp action is defined by a tuple constituting the grasp position, grasp orientation and the placement position as $a^{grasp} = \{\mathbf{p}^{grasp}, \alpha^{grasp}, \mathbf{p}^{place}\}$. The grasp affordances are generated using the GG-CNN framework (Morrison et al., 2020). The network takes as

input the depth image and provides as output the grasp position $\mathbf{p}^{grasp}$, grasp orientation α^{grasp} and grasp confidence measure q_k. The grasp quality measure is used for providing the edge attributes to the declutter scene graph. The placement position $\mathbf{p}^{place}$ is defined at a predetermined position away from the clutter. In order to get object specific grasp measures, the masked depth input was provided to the grasp affordance network using the semantic segmentation outputs. The grasp affordance network depends on the quality of the depth images provided as input and hence provides noisy grasp affordances when used from a static viewpoint. In order to improve the grasp estimates, the robot is autonomously moved to a viewpoint at a predefined height over the object to grasp using the centroid information from the semantic segmentation output. The grasp action is considered to be successful when the object is correctly grasped and placed at the desired location within a predefined tolerance of 5 cm. The tactile sensors on the gripper are monitored for loss of contact during grasp and place operation. If the robot fails to grasp at any point and drops the object, it is detected by the tactile sensors and the action is repeated. A maximum repetition counter of 5 times was set after which a human intervened to reset the complete scene. The grasp action is illustrated in Fig. 5.4a.

Push Action for Decluttering

The push action is parameterized by a tuple with the push point and push direction $a^{push} = (\mathbf{p}^{push}, \overrightarrow{\mathbf{d}}^{push})$. The push trajectory is a straight line for a pre-defined distance. Quasi-static pushing (Mason, 1986b) on a flat surface with uniform friction between the object and the sliding surface is assumed. Using the instance segmentation mask of the object to be pushed, vectors $\mathbf{v}_{i,k} \forall i$ are computed between the centroid of the bounding box of the pushed object and all the other objects. The vector pointing towards the clutter is provided by $\mathbf{v} = \sum_i w_i \mathbf{v}_{i,k}$. Consequently, each vector is weighted with scalar weights w_i that are inversely proportional to the distance such that higher weights are assigned to objects closer to the pushed object. The final push direction is provided by $\overrightarrow{\mathbf{d}}^{push} = -\frac{\mathbf{v}}{||\mathbf{v}||}$ as shown in Fig. 5.3b. The push point p^{push} is calculated as the point at the intersection of the contour of the segmentation mask and push direction $\overrightarrow{\mathbf{d}}^{push}$ placed at the centroid. This ensures the push action is acting along the centroid of the object. However, as the width of the fingertips of the gripper is bulky (about 3 cm), it may not be always possible to ensure the robot to reach the position p^{push} without colliding with other objects in the clutter. Hence, points are sampled along the contour of the instance segmentation mask of the object and the bounding box of the fingertip size is projected on these sampled points in the image. The bounding box projection is calculated through the camera intrinsics and the extrinsic hand-eye calibration matrix. The mean Intersection-over-Union (IoU) of

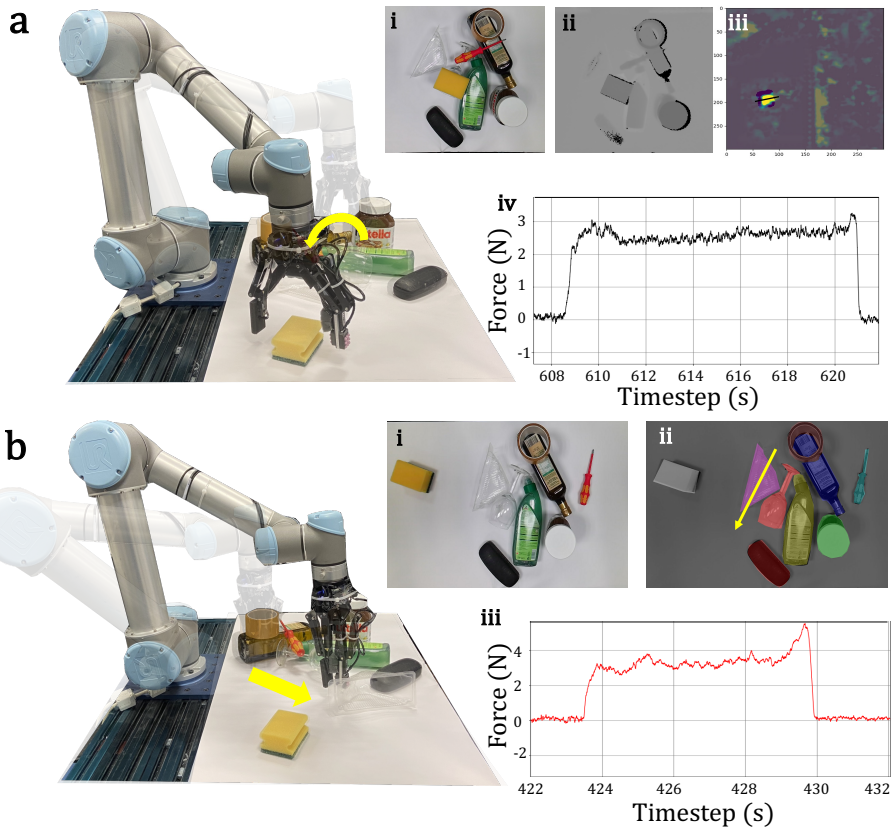


Figure 5.4: a) Grasp action performed by the robot, i) RGB input, ii) depth input, iii) grasp affordance output, iv) tactile signal values during the grasp action. b) Push action performed by the robot, i) RGB input, ii) Semantic segmentation and push affordance output, iii) tactile signal values during the push action

the gripper fingertip bounding box is calculated with the other objects and the sampled point resulting in the least mean *IoU* value is chosen as $\mathbf{p}^{push}$. The action is performed successfully if the object is pushed to the desired position within a certain tolerance of 5 cm. As unintended object movements are expected during pushing due to the varying shapes and center of mass positions of the objects, the tactile sensors on the grippers are monitored to detect any loss of contact. This triggers a recalculation of the push action and the process is repeated again. A maximum repetition counter for 5 times is set after which a human intervenes and resets the complete scene. The push action process is shown in Fig. 5.4b.

5.2.3 Shared Visuo-Tactile based Active Object Reconstruction

As the shape of the target object is unknown, reconstruction of the object is necessary for pose estimation and other possible downstream tasks such as manipulation. The frame-

work autonomously chooses (a) which sensor to use, (b) where to perform sensing and (c) how much of the object information is necessary for the chosen objective of pose estimation. In a single sensor scenario, the *next best action selection* problem seeks to find the optimal next sensory action to perform based on current knowledge of the environment in order to maximise the information gain that is calculated through an objective function. In a multi-agent and multi-sensor scenario, there is additionally the *sensor selection* problem which seeks the find the optimal sensor to employ given the current knowledge of the environment and incentivises the coordination between the agents as well as reducing the redundant data collection. The two robots equipped with a visual RGB-D sensor and tactile sensor array respectively as shown in Fig. 5.1 are tasked to reconstruct the object in a coordinated and time-efficient manner.

Vision and Tactile Action Sampling: For the Next-Best-View (NBV) and Next-Best-Touch (NBT) selection, Monte-Carlo sampling of the visual and tactile actions respectively is performed around the target object. The centroid $o_{centroid}$ of the target object is extracted from the semantic segmentation mask.

For NBV sampling, N_{nbv} viewpoints are sampled on the hemisphere space centered on $o_{centroid}$ of the target object. A viewpoint $a^v \in \mathcal{A}^v$ is defined by the position $\mathbf{p}^v \in \mathbb{R}^3$ and orientation $\mathbf{R}^v \in SO(3)$ of the camera frame expressed in the world coordinate frame ${}^wH_c$. The constraint on a^v is that the camera orientation must be towards the object of interest i.e., the Z-axis of the camera frame which points outward from the camera needs to pass through the centroid of the target object. The position $\mathbf{p}^v$ is randomly sampled as a point on the hemisphere (Marsaglia, 1972). The points which are sampled outside the kinematic limits of the robot are discarded. The rotation matrix $\mathbf{R}^v$ is calculated through the angle-axis formulation $\{\hat{e}, \theta\}$ as follows:

$$\hat{\mathbf{h}} = \frac{\mathbf{p}^{view} - \mathbf{o}_{centroid}}{||\mathbf{p}^{view} - \mathbf{o}_{centroid}||} \tag{5.3}$$

$$\theta = \cos^{-1}(\hat{\mathbf{h}} \cdot \hat{\mathbf{Z}}), \quad \hat{\mathbf{e}} = \frac{\hat{\mathbf{h}} \times \hat{\mathbf{Z}}}{||\hat{\mathbf{h}} \times \hat{\mathbf{Z}}||} \tag{5.4}$$

where $\hat{\mathbf{Z}} = \{0,0,1\}$ the Z-axis of the world coordinate frame $\mathcal{W}$. The rotation matrix $\mathbf{R}^v$ can be calculated from the angle-axis formulation using the Rodrigues' formula (Murray et al., 2017) as follows:

$$\mathbf{R}^v = \begin{bmatrix} \cos\theta + \hat{\mathbf{e}}_x^2(1-\cos\theta) & \hat{\mathbf{e}}_x\hat{\mathbf{e}}_y(1-\cos\theta) - \hat{\mathbf{e}}_z\sin\theta & \hat{\mathbf{e}}_y\sin\theta + \hat{\mathbf{e}}_x\hat{\mathbf{e}}_z(1-\cos\theta) \\ \hat{\mathbf{e}}_z\sin\theta + \hat{\mathbf{e}}_x\hat{\mathbf{e}}_y(1-\cos\theta) & \cos\theta + \hat{\mathbf{e}}_y^2(1-\cos\theta) & -\hat{\mathbf{e}}_x\sin\theta + \hat{\mathbf{e}}_y\hat{\mathbf{e}}_z(1-\cos\theta) \\ -\hat{\mathbf{e}}_y\sin\theta + \hat{\mathbf{e}}_x\hat{\mathbf{e}}_z(1-\cos\theta) & \hat{\mathbf{e}}_x\sin\theta + \hat{\mathbf{e}}_y\hat{\mathbf{e}}_z(1-\cos\theta) & \cos\theta + \hat{\mathbf{e}}_z^2(1-\cos\theta) \end{bmatrix} \tag{5.5}$$

where $\hat{\mathbf{e}} = \{\hat{\mathbf{e}}_x, \hat{\mathbf{e}}_y, \hat{\mathbf{e}}_z\}$. For the NBT sampling, the tactile action is defined as $a^t \in \mathcal{A}^t$ by the position $\mathbf{p}^t \in \mathbb{R}^3$ and direction $\hat{d}$. The positions are randomly sampled as points on each face of the oriented bounding box of the object except the bottom face where the object rests on the table. The bounding box is determined from the 2D semantic segmentation mask of the target object and a predefined height. The direction $\hat{d}$ of each $\mathbf{p}^t$ is calculated as the normal perpendicular to the face of the bounding box. The NBV and NBT sampling are graphically shown in Fig. 5.5.

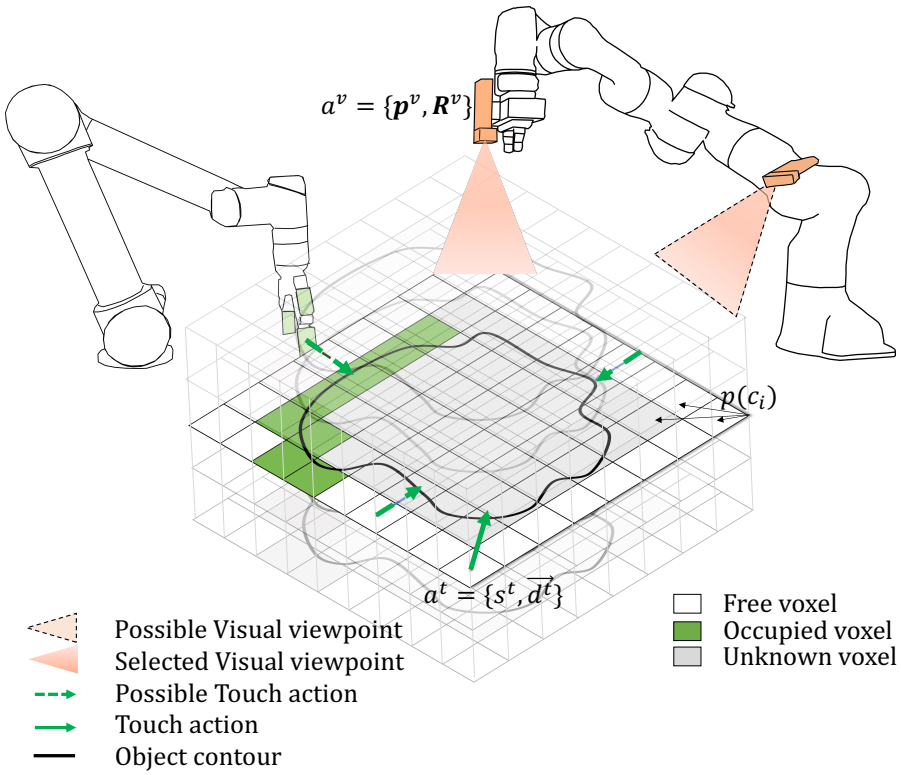


Figure 5.5: Next best view (NBV) and next best touch (NBT) action selection

Active Sensor Selection and Next Best Action Selection: At each iteration, the next best action a^* is selected from the set $\mathcal{A} = \mathcal{A}^v \bigcup \mathcal{A}^t$ using the joint information gain approach. The space around the target object is discretised into a 3D voxelised probabilistic occupancy grid G with resolution G_{res}. Each grid cell $g_i \in G$ is represented by a Bernoulli random variable X_g which represents the probability if the grid cell is occupied $X_g = 1$

or unoccupied $X_g = 0$. These Bernoulli random variables are assumed to be independent which allows the calculation of the probabilities of the occupancy grid $p(g_i)$. The confidence of the reconstruction can be calculated as the uncertainty of the grid through the Shannon Entropy as:

$$\mathbb{H}(G) = -\sum_{g_i \in G} p(g_i) log(p(g_i)) + (1 - p(g_i)) log(1 - p(g_i)) \tag{5.6}$$

Given a point cloud captured by the camera or tactile sensor, the occupancy grid is updated with probabilities using the respective sensor models (Hornung et al., 2013). A virtual sensor measurement model for visual and tactile sensors are defined for the NBV and NBT calculations. The visual sensor generates the point cloud using a time-of-flight (ToF) sensor. The virtual vision sensor model is defined by a set of beam measurements $R_v = r_1, r_2, r_3, \ldots r_{n_v}$ where r_{n_v} refers to the maximum number of rays. These rays are rotated such that they span the field-of-view of the sensor. Similarly, a virtual tactile sensor is defined which casts a set of rays $R_t = r_1, r_2, r_3, \ldots r_{taxel}$ where r_{taxel} refers to the number of taxels in the sensor array. Raytracing is used to update the grid cells, wherein the grid cells where the ray terminates is updated as hits and remaining grid cells are updated as misses. Given the observed grid cell g and the measurement from sensor observation z, the log-odds is updated as $L(g|z) = L(g) + l(z)$ wherein $L(g) = log \frac{p(g)}{1-p(g)}$ and

$$l(z) = \begin{cases} log \frac{p_h}{1-p_h} & z \widehat{=} \, hit \\ log \frac{p_m}{1-p_m} & z \widehat{=} \, miss \end{cases} \tag{5.7}$$

where p_h and p_m are the probabilities of hit and miss which are user-defined values set to 0.7 and 0.4 respectively as in (Hornung et al., 2013). The posterior probability $p(g|z)$ can be computed by inverting $L(g|z)$.

For a single sensor case, the expected information gain by taking an action $a_k \in \mathcal{A}$ and corresponding expected measurement $\hat{z}_t$ is given by the Kullback–Leibler (KL) divergence between the posterior entropy after integrating the expected measurements and the prior entropy:

$$E[\mathbb{I}(p(g_i|a_t, \hat{z}_t))] = \mathbb{H}(p(g_i)) - \mathbb{H}(p(g_i|a_t, \hat{z}_t)) \tag{5.8}$$

Hence, the action a^* that maximises the information gain can be selected as the next best action:

$$a^* = \underset{a_k \in \mathcal{A}}{\arg\max} (E[\mathbb{I}(p(g_i|a_k, \hat{z}_k))]) \tag{5.9}$$

A näive way of extending to multiple sensors is to compute Eq. (5.9) for each sensor.

However, this would result in collection of redundant data due to sensor data overlap. Furthermore, there would be no incentive for coordination between the robots as well as leveraging the vision and tactile sensors with complementary properties. Hence, a joint sensor selection and action selection method is proposed for vision and tactile sensors. The *same* occupancy grid formulation can be utilised for integrating the sensor information from vision and tactile sensors. For each cell of the occupancy grid, the probabilistic evidence from the each sensor needs to be updated. An energy cost $D(a_t)$ is defined that encodes the time taken to perform the robot action. In general, performing visual actions is faster than performing tactile actions with the robot. Hence, the energy cost is set as $D(a_t^v) < D(a_t^t)$ for all cases unless the target object is transparent. In case of objects that are transparent, the visual sensor produces erroneous data and the energy cost was set as $D(a_t^t) < D(a_t^v)$, thus preferring the tactile actions in such cases. The proposed method for detection of transparent objects is described in following subsection. Hence the optimization for the sensor selection and next best action is performed by:

$$a^* = \arg\max_{a_k \in \mathcal{A}} \left(\frac{E[\mathbb{I}(p(g_i|a_k, \hat{z}_k))]}{D(a_k)} \right) \tag{5.10}$$

Detection of Transparent Objects: Detecting transparent objects is a challenging tasks for off-the-shelf visual cameras with RGB and depth sensing. Many prior works are available for detection of transparent objects with the usage of specialised sensors or specific calibration setups, and with analytical or data-driven methods (Ihrke et al., 2010). A simple heuristic approach is proposed to detect object transparency in order to set the

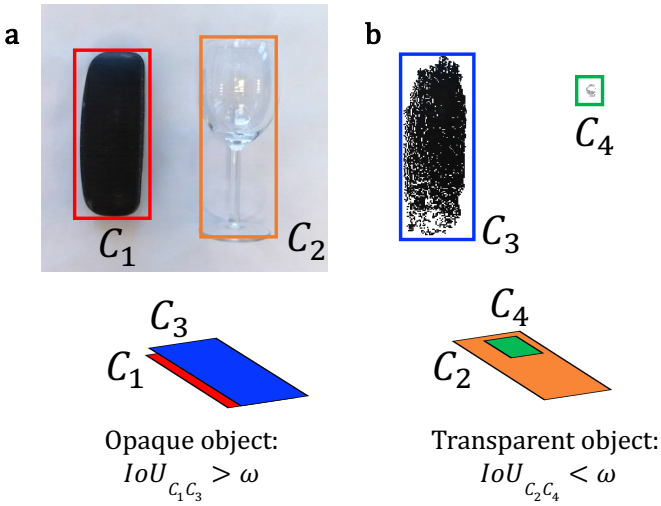


Figure 5.6: Bounding box segmentation and IoU calculation using (a) RGB image (b) Point cloud for detecting transparent objects.

Algorithm 2: Shared Visuo-Tactile Object Exploration

Input: Current point cloud P
Result: Next best action a^*
Action Sampling:
$\mathcal{A}^v \leftarrow \text{sample}(\mathbf{p}^v, \mathbf{R}^v)$
$\mathcal{A}^t \leftarrow \text{sample}(\mathbf{p}^t, \hat{d})$
$\mathcal{A} = \mathcal{A}^v \cup \mathcal{A}^t$
Next Best Action Selection:
Update prior entropy of occupancy grid:
$\mathbb{H}(G) \leftarrow$ prior entropy ; ▷ Eq. (5.6)
Extract virtual measurements: $\hat{z}_k \; \forall a_k \in \mathcal{A}$
Update posterior entropy of occupancy grid:
$\mathbb{H}(G|\hat{z}) \leftarrow$ posterior entropy
Calculate information gain:
$E[\mathbb{I}(p(g_k|a_k, \hat{z}_k))]$; ▷ Eq. (5.8)
Set $D(a)$ given type of target object
Calculate next best action:
$a^* = \arg\max_{a \in \mathcal{A}} \frac{E[\mathbb{I}(p(g_k|a_k, \hat{z}_k))]}{D(a)}$; ▷ Eq. (5.10)

energy cost $D(a_t)$ during object exploration. The RGB image and point cloud of the target object are extracted from a perpendicular top-down view. The bounding box $\mathcal{C}_{rgb}$ of the object is extracted from the RGB image using contour segmentation techniques (Bradski and Kaehler, 2000). Assuming that the object lies on a plane, plane segmentation techniques were employed to remove the points belonging to the plane from the point cloud. The 2D bounding box of the remaining points (ignoring the height of the bounding box) is extracted as $\mathcal{C}_{pc}$. The overlap measure between $\mathcal{C}_{rgb}$ and $\mathcal{C}_{pc}$ which is measured by the Intersection-over-Union (IoU), as previously defined in Sec. 5.2.2, is used classify the object transparent if $IoU_{pc/rgb} < \omega$ or opaque. The transparent object detection strategy is visualised in Fig. 5.6. The detected transparent object is used to set the value for $D(a_t^t)$ such that it is less than $D(a_t^v)$. This ensures that tactile actions are preferred for object exploration in case of transparent objects. The algorithm for shared visuo-tactile object exploration is summarized in Algorithm 2.

Category-level Object Shape Reconstruction

Category-level reconstruction refers to the problem setting where the exact instance of the object model is not available but rather the category to which the object belongs is known a priori. In order to recognise the shape of category-level objects, a self-supervised learning approach with an autoencoder network is presented that aims to reconstruct the

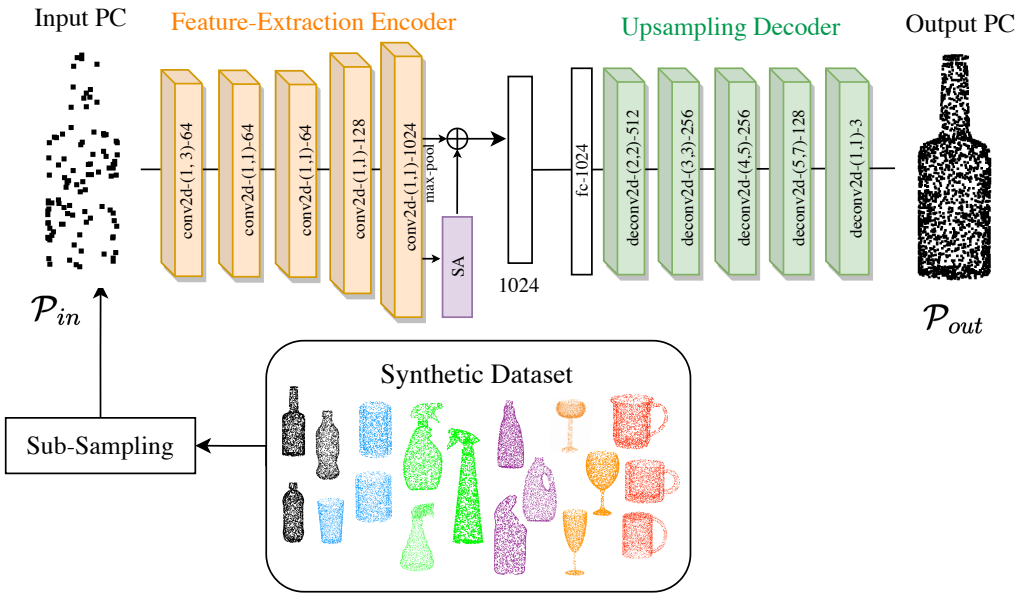


Figure 5.7: Architecture for the reconstruction network.

original point cloud when provided a subsampled point cloud. The network is trained on only *synthetic object models* belonging to the same category but not identical as the real-world objects. A dataset $\mathcal{D}$ of synthetic point clouds was generated from synthetic CAD models available in the ShapeNet repository (Chang et al., 2015). The trained network is directly used with visual and tactile point clouds from real-world objects. This avoids expensive real-world data collection and annotation process. The network consists of a feature-extraction encoder and upsampling decoder unit as shown in Fig. 5.7. The input point clouds $\mathcal{P}_{in}$ are randomly sampled with different sampling factors to produce point clouds with point numbers between 60 and 1024. The reconstructed output point cloud $\mathcal{P}_{out}$ from the network is fixed to 2048 points. This is done to emulate tactile-only and visuo-tactile point clouds which are sparse and dense respectively. The low density of the input point clouds provides a challenge for shape reconstruction as other simpler techniques such as interpolation cannot be used. The trained network is used for inference with sensor acquired point clouds without any fine-tuning with real-data. Given that the reconstruction methodology is comprehensively detailed in Chap. 8, an in-depth discussion is omitted in this section for brevity.

5.2.4 Visuo-Tactile based Robust Pose Estimation

Stochastic Translation-Invariant Quaternion Filter (S-TIQF)

Following the reconstruction of the object point cloud, pose estimation is carried out to determine the unknown scale $\mathbf{S} \in \mathbb{R}^3$, rotation $\mathbf{R} \in SO(3)$, and translation $\mathbf{t} \in \mathbb{R}^3$. As defined in Eq. (4.1), the point cloud registration problem given any two point clouds ($\mathcal{S}$ and $\mathcal{O}$) with known point-to-point correspondences is given as:

$$\mathbf{s}_i = \mathbf{S}.(\mathbf{R}\mathbf{o}_i) + \mathbf{t} \quad i = 1, \dots N \tag{5.11}$$

where $\mathbf{s}_i \in \mathbb{R}^3$ are points in the point cloud $\mathcal{S}$ and $\mathbf{o}_i \in \mathbb{R}^3$ are the corresponding points belonging to the point cloud $\mathcal{O}$ and . represents the element-wise product. Typically, for object pose estimation the point cloud $\mathcal{S}$ is derived from scene with sensor measurements and the point cloud $\mathcal{O}$ is derived from the object model. Unlike in Chap. 4 wherein the exact object CAD model is known, in this case, the reconstructed object model is used from the Sec. 5.2.3 as $\mathcal{O}$. Hence the scaling factor needs to be estimated.

Scale Estimation: The reconstructed object point cloud $\mathcal{O}$ is uniformly scaled within a $[0,1]^3$ cube. To find the absolute scale $\mathbf{S}$, the ratio of the axis aligned bounding box (AABB) of the scene $\mathcal{S}$ and object $\mathcal{O}$ point clouds is computed, i.e., if $\{(x_{min}, x_{max}), (y_{min}, y_{max})$ $(z_{min}, z_{max})\}$ represents the AABB for a point cloud, then:

$$\mathbf{S} = \{\frac{|x_{max} - x_{min}|_{\mathcal{S}}}{|x_{max} - x_{min}|_{\mathcal{O}}}, \frac{|y_{max} - y_{min}|_{\mathcal{S}}}{|y_{max} - y_{min}|_{\mathcal{O}}}, \frac{|z_{max} - z_{min}|_{\mathcal{S}}}{|z_{max} - z_{min}|_{\mathcal{O}}}\} \tag{5.12}$$

Subsequently, the object point cloud $\mathcal{O}$ can be scaled by $\mathbf{S}$ and the estimation of $\mathbf{R}$ and $\mathbf{t}$ remains. Eq. (5.11) simplifies to $\mathbf{s}_i = \mathbf{R}\mathbf{o}'_i + \mathbf{t}$ where $\mathbf{o}'_i = \mathbf{S}\mathbf{o}_i$ for $i = 1, 2, \dots N$ which is the same as Eq. (4.1) in Chap. 4.

Stochastic Initialization: The Translation-Invariant Quaternion Filter (TIQF) is presented in the Chap. 4 which is a Kalman filter approach for point cloud registration applicable for dense visual and sparse tactile point clouds. However, TIQF is sensitive to initialisation conditions. Fig. 5.8 shows an example error surface for point cloud registration with TIQF using the Stanford Bunny dataset (Levoy et al., 2005). It is obtained by varying the initial position about one axis in the range of $[-5.0, 5.0]$ and initial orientation about one axis in the range of $[-\pi, \pi]$. The error is calculated as the root mean squared error of the distance metric between corresponding points. It can be noted from Fig. 5.8 that the error surface contains multiple local minima in which the optimization can be trapped depending upon the initial conditions. This problem is solved with a stochastic initialization

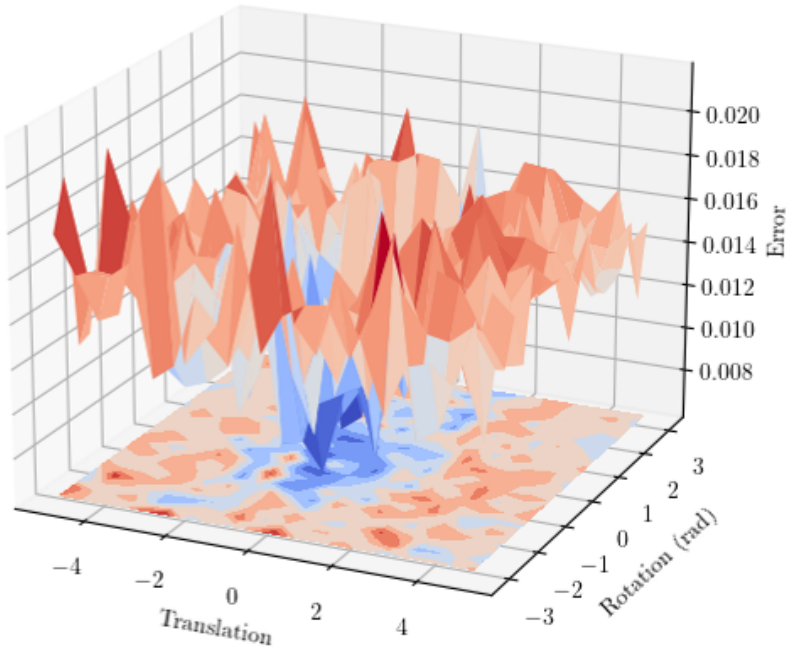


Figure 5.8: Error surface calculated as the distance between corresponding points of two clouds upon performing TIQF with initialisation parameters (translation and rotation) varied (best viewed on-screen and in colour).

method for TIQF that is robust against local optima termed StochasTIQF (S-TIQF).

The stochastic alignment is performed through Simulated Annealing (Bertsimas and Tsitsiklis, 1993). Simulated Annealing (SA) is a well-known stochastic probabilistic method for approximating the global optima for a given function $f(b)$. Simulated Annealing was chosen for the following reasons: (a) very effective to escape local minima (Granville et al., 1994), (b) low number of parameters to tune, (c) cost function for SA in this application (cf. Eq. 5.14) has clear geometric interpretations and (d) simpler to implement compared to other heuristic optimisation algorithms (Henderson et al., 2003). In SA, a temperature variable is used to guide the exploration. An initial temperature $t = t_0$ was chosen. The annealing schedule was chosen as the geometric progression as: $t' = t\zeta$ where ζ is the cooling rate. In the experiments, the cooling rate was set as $\zeta = 0.98$. At $t = t_0$, an initial state $b = b_0$ is chosen at random and the cost is computed using the cost function $c_0 \leftarrow f(b_0)$. At every iteration, a random state in the neighbourhood of the current state is chosen and the difference in cost Δc is calculated. The probability of accepting the new state is provided by the following condition:

$$p(b') = \begin{cases} 1 & \Delta c \leq 0 \\ e^{\frac{-\Delta c}{t}} & \Delta c > 0 \end{cases} \tag{5.13}$$

The new state b' is accepted if $p(b') > \text{random}(0,1)$. The process is repeated until a predefined temperature threshold is reached $t < t_{min}$ or for a fixed number of iterations. Random restarts are also used wherein t is set to t_0 when $t \leq t_{min}$. In the experiments, random restarts were performed 10 times. In the experiments, the parameters were set as follows: $t_0 = 1$ and $t_{min} = 0.0001$. In order to use Simulated Annealing with TIQF, a cost function for SA needs to be designed that upon finding the solution provides a good initialization for TIQF to extract the rotation and translation estimates. The cost is defined as the root mean squared error of the nearest neighbour point-to-point distances for each state $b = \{R,t\}$. The nearest neighbourhood correspondence assignment allows fast computation of the costs and thereby allowing larger iterations of SA. The state with minimal cost naturally minimizes the distance between the two point clouds. Hence for two point sets $\mathcal{S}$ and $\mathcal{O}$, the cost is defined as follows:

$$f(b) = \frac{1}{|\mathcal{O}|}(\sum \sqrt{||\min(\mathcal{S} - (\mathbf{R}_{SA}\mathcal{O} + \mathbf{t}_{SA}))||^2}) \tag{5.14}$$

The temperature variable allows exploration in the initial phase thereby escaping the local minima and gradually converges to an optimal solution. The estimated rotation $\mathbf{R}_{SA}$ and translation $\mathbf{t}_{SA}$ is used for TIQF to find the accurate solution.

Correspondence Estimation: A crucial factor in point cloud registration from Eq. (5.11) is the knowledge of point correspondences in the two point sets. In realistic scenarios, the point correspondences are not known *a priori*. On the one hand, simultaneous pose and correspondence estimation methods such as ICP and its variants rely upon nearest neighbour search for extracting point correspondences while iteratively improving the pose in successive steps (Besl and McKay, 1992). On the other hand, correspondence-based methods extract point correspondences through feature matching and may employ rejection techniques to remove outlier correspondences prior to performing registration. In the case of visual and tactile point clouds, there are further challenges: (a) the point density difference between visual and tactile point clouds and (b) visual point clouds can be captured in one shot whereas tactile point cloud is aggregated through sequential tactile actions. Due to the point sparsity, typical feature-based correspondence matching algorithms are not accurate as they depend on local surface information. Similarly, nearest neighbour search as used in ICP is not robust to outliers and can get stuck in local minima.

The mutual nearest neighbours or *Best-Buddies Pairs (BBP)* (Oron et al., 2017) method were used to estimate the point correspondences. It has been shown in (Oron et al., 2017) that the BBP measure is robust to outliers and difference in point density but in the context of template matching in the image domain. The point $p_i \in \mathcal{P}$ and $q_j \in \mathcal{Q}$ are *Best Buddy*

Pairs (BBP) if p_i is the nearest neighbour of q_j in point cloud $\mathcal{Q}$ and q_j is the nearest neighbour of p_i in point cloud $\mathcal{P}$. Mathematically, it can be written as:

$$bbp(p,q,\mathcal{P},\mathcal{Q}) = \begin{cases} 1 & NN(p_i,\mathcal{Q}) = q_j \wedge NN(q_i,\mathcal{Q}) = p_j \\ 0 & \text{otherwise} \end{cases} \tag{5.15}$$

where $NN(p_i,\mathcal{Q}) = \arg\min_{q\in\mathcal{Q}} d(p_i,q)$ and $d(p_i,q)$ is a distance measure. Typically, the nearest neighbours can be calculated based on Euclidean distances as $d(p_i,q) = ||q - (\hat{\mathbf{R}}p_i + \hat{\mathbf{t}})||$ where $\hat{\mathbf{R}}$ and $\hat{\mathbf{t}}$ are the current rotation and translation estimates respectively. Furthermore, the normals are also included in case there are multiple candidates for nearest neighbours. The corresponding points p_i and q_j must also have their associated normals n_{p_i} and n_{q_j} oriented approximately in similar directions i.e., the $NN(p_i,q_j) = 1$ if $\arccos(n_{q_j}.\hat{\mathbf{R}}n_{p_j})$ is less than an user-defined threshold.

Rotation and Translation Estimation: The estimation of rotation and translation is decoupled and performed in consecutive steps as provided in Chap. 4 (Sec. 4.2.3). The decoupling is done by computing the relative vectors between pairs of corresponding points as $\mathbf{s}_{ji} = \mathbf{s}_j - \mathbf{s}_i$ and $\mathbf{o'}_{ji} = \mathbf{o'}_j - \mathbf{o'}_i$. The Eq. (5.11) is simplified as:

$$\mathbf{s}_j - \mathbf{s}_i = (\mathbf{R}\mathbf{o'}_j + \mathbf{t}) - (\mathbf{R}\mathbf{o'}_i + \mathbf{t}) \tag{5.16}$$

$$\mathbf{s}_{ji} = \mathbf{R}\mathbf{o'}_{ji} \tag{5.17}$$

The Eq. (5.17) is independent of translation $\mathbf{t}$, hence these measurements are termed as *translation-invariant measurements*. The rotation estimation is done as detailed in Sec. 4.2.3 in Chap. 4 for TIQF. Once the rotation estimate $\mathbf{R}$ is found, the translation estimate $\mathbf{t}$ is computed in closed form:

$$\mathbf{t} = \frac{1}{N}\sum_{i=0}^{N}(\mathbf{s}_i - \mathbf{R}\mathbf{o'}_i) \tag{5.18}$$

Thus, with each iteration of the S-TIQF, a new rotation and translation estimate are obtained which are used to transform the model. The transformed model is used to recompute correspondences and repeat the S-TIQF update steps. The change in homogeneous transformation between iterations $\Delta_{TIQF} < \xi^{conv}$ is calculated for convergence criteria i.e., if the difference in the output pose is less than a specified threshold which in the experiments is 0.1 mm and 0.1° respectively and/or maximum number of iterations in order to check for convergence ($max\_it_{S-TIQF} = 100$). The psuedo-code of the S-TIQF algorithm is shown in Algorithm 3.

Algorithm 3: S-TIQF

Input: Point clouds $\mathcal{S}$, $\mathcal{O}$
Result: Estimated rotation **R**, translation **t**, scale **S**
Scale Estimation:
$AABB_S = \texttt{calculate\_AABB}(\mathcal{S})$, $AABB_O = \texttt{calculate\_AABB}(\mathcal{O})$,
$\mathbf{S} = \frac{AABB_S}{AABB_O}$; ▷ Eq. (5.12)
$\mathcal{O} \leftarrow \texttt{scalePC}(\mathbf{S})$
Stochastic Initial Alignment:
Initialisation: $t = t_{init}, \zeta = 0.98, t_{min} = 10^{-5}, T_{SA} = \mathbb{I}^{4\times4}$
$c_{best} = \texttt{getCost}(\mathcal{O}, \mathcal{S}, \texttt{T}_{\texttt{SA}})$; ▷ Eq. (5.14)
while $\texttt{restarts} < \texttt{max\_restarts}$ **do**
  while $t > t_{min}$ **do**
    $t \leftarrow t\zeta$; $T_{rand} \leftarrow \texttt{getRandomTransformation}()$
    $c \leftarrow \texttt{getCost}(\mathcal{O}, \mathcal{S}, \texttt{T}_{\texttt{rand}})$; ▷ Eq. (5.14)
    $T_{SA} \leftarrow \texttt{acceptanceProbability}(\texttt{c}, \texttt{c}_{\texttt{best}})$; ▷ Eq. (5.13)
  end
  $t \leftarrow t_{init}$
  $\texttt{restarts}++$
end
$\mathcal{O} \leftarrow \texttt{transformPC}(\texttt{T}_{\texttt{SA}})$
TIQF:
while $\neg\texttt{converged}$ **do**
  $s_i \in \mathcal{S}, o_i \in \mathcal{O} \leftarrow \texttt{correspondenceEstimation}(\mathcal{S}, \mathcal{O})$ $s_{ji}, o_{ji} \leftarrow \texttt{TIMS}(\texttt{s}_\texttt{i}, \texttt{o}_\texttt{i})$;
  ▷ Eq. (5.17)
  $\mathbf{R} \leftarrow \texttt{rotationEstimation}(\texttt{s}_{\texttt{ji}}, \texttt{o}_{\texttt{ji}})$; ▷ Sec. 5.2.4
  $\mathbf{t} \leftarrow \texttt{translationEstimation}(\texttt{s}_{\texttt{ji}}, \texttt{o}_{\texttt{ji}})$ $\mathcal{O} \leftarrow \texttt{transformPC}(\mathbf{R}, \mathbf{t})$
  $\texttt{converged} \leftarrow \texttt{convergenceCriteria}()$
end

5.2.5 Visuo-Tactile Hand-Eye Calibration

As described in the framework in Fig. 5.2bi, if there is a discrepancy between visual and tactile point clouds for a static object, it is typically due to incorrect hand-eye calibration (c.f., Fig. 5.20a). Conventionally, the hand-eye calibration is performed using a specialized target such as a calibration grid as shown in Fig. 5.9a. However, this process is time-consuming as it adds additional overhead such as specialized targets and calibration procedures. Furthermore, the grid-based calibration technique may contain residual errors as it is dependent on the chosen robot end-effector poses, lighting conditions, and sensor noise. Recent works have introduced deep-learning based markerless hand-eye calibration methods using segmentation and differentiable rendering techniques to regress the camera-to-robot pose based on input images of the robot and associated joint kinemat-

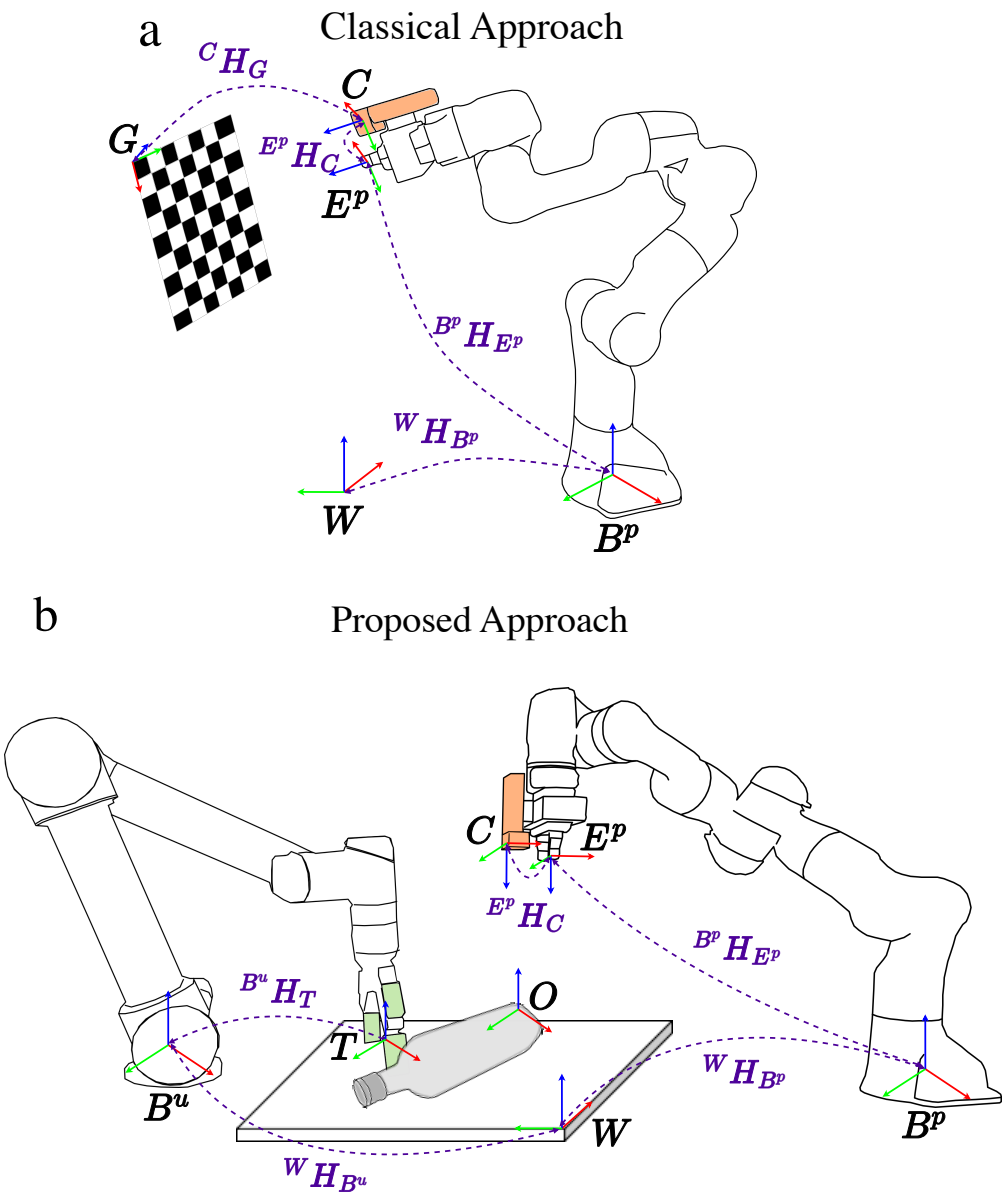


Figure 5.9: (a) Classical grid-based hand-eye calibration method (b) Proposed in-situ visuo-tactile hand-eye calibration method

ics (Lu et al., 2023; Labbé et al., 2021). While the disadvantages of solely relying upon visual images such as occlusions, challenging backgrounds for segmentation, and lighting conditions still apply, there is an additional overhead of training requirement for various types and kinematic configurations of the robots. Furthermore, these methods can regress the camera-to-robot pose only in cases wherein the robot is visible to the camera but cannot be used in eye-in-hand cases wherein the camera is attached to the end-effector of the robot. In this section, the need for a specific calibration artifact or target has been re-

laxed and hand-eye calibration has been performed using any known arbitrary object in the workspace of the robot, hence termed as *in-situ* calibration. Moreover, integrating visual and tactile modalities substantiates the estimation, refining the visual approximations through the incorporation of sparse tactile measurements, thereby enhancing the precision of the hand-eye calibration.

Consider the two-manipulator system shown in Fig. 5.9b. The hand-eye calibration problem of finding ${}^{E^P}H_C$ is casted as a point cloud registration problem given any arbitrary object. The camera frame C can be expressed in the world coordinate frame W as follows:

$$ {}^{W}H_C = {}^{W}H_{B^P}\ {}^{B^P}H_{E^P}\ {}^{E^P}H_C \tag{5.19} $$

where ${}^{W}H_{B^P}$ is known *a priori* through user assignment of the world frame and ${}^{B^P}H_{E^P}$ is extracted through the robot kinematic model and ${}^{E^P}H_C$ is the so-called hand-eye calibration matrix. The coordinate frames B^P and E^P are the base frame and end-effector frame of the Panda robot respectively and B^U is the base frame of the UR5 robot. Therefore, the hand-eye calibration matrix can be obtained as:

$$ {}^{E^P}H_C = {}^{W}H_{E^P}^{-1}\ {}^{W}H_C \tag{5.20} $$

Let's denote ${}^{W}\hat{H}_C$ as the estimated ${}^{W}H_C$. The transformation ${}^{W}H_C$ can be estimated through point-set registration that minimises the following cost function:

$$ f({}^{W}\hat{H}_C, {}^{C}P_i^v, {}^{W}P_i^t) = \frac{1}{N}\sum_{i=1}^{N} ||{}^{W}\hat{H}_C\,{}^{C}P_i^v - {}^{W}P_i^t||^2 \tag{5.21} $$

where ${}^{C}P_i^v$ is the point cloud of an arbitrary object captured by the camera in the camera frame and ${}^{W}P_i^t$ is the point cloud extracted using tactile sensing of the same arbitrary object in the world coordinate frame and $i = 1,2,3\ldots N$ represent the corresponding points in the two aforementioned point clouds. The position of the arbitrary calibration object is rigidly fixed to the workspace such that it does not move during tactile probing actions. Furthermore, as guarded motions were used for tactile probing actions where the force values on the tactile sensors were monitored and the robot stopped immediately upon contact, thus ensuring the object did not move while performing tactile actions. The tactile point cloud is expressed in the world frame through the following transformations:

$$ {}^{W}P^t = {}^{W}H_{B^u}\,{}^{B^u}H_T\,{}^{T}P^t \tag{5.22} $$

where the coordinate frame T is the frame of the tactile sensor. The transform ${}^{W}H_{B^u}$ is

defined *a priori* by the user assignment of the world frame and as the tactile sensors are rigidly attached to the gripper, the corresponding transform ${}^{B^u}H_T$ is received from the robot kinematic model. The transformation ${}^{B^u}H_T$ can be obtained from the kinematic model of the robot and any miscalibration of the tactile sensor frame due to possible incorrect mounting of the tactile sensors are absent. Other scenarios may involve the tactile sensors and the visual sensors attached to the same robot. The formulation involving a multi-robot setup can be simplified trivially for a single robot setup as well.

As the tactile data is high fidelity, the aim is to register the dense visual point cloud ${}^{C}P^{v}$ to the sparse tactile point cloud ${}^{W}P^{t}$ using the S-TIQF algorithm as detailed in Section 5.2.4. Note that any point cloud registration method can be used but as it is demonstrated in Sec. 5.3, state-of-the-art point cloud registration methods perform poorly in dense-sparse registration whereas the S-TIQF approach shows high accuracy even with low number of points. The S-TIQF algorithm produces the homogeneous transform ${}^{W}\hat{H}_C$ as output. Plugging the ${}^{W}\hat{H}_C$ value into Eq. (5.20), the required hand-eye calibration matrix ${}^{E^P}H_C$ can be obtained.

5.3 Experimental Results

5.3.1 Experimental Setup

The experimental setup shown in Fig. 5.1 consists of a Universal Robots UR5 robot with a tactile sensorised Robotiq 2F140 Gripper and Franka Emika Panda robot with the standard Panda Gripper. The tactile sensor array of the two-finger gripper are acquired from Xela robotics©and Contactile©as explained in Chap. 3. Each taxel of both types of the sensor arrays provides 3-axis force measurements. This configuration allows the robot to acquire tactile data while touching with the outer side and from the fingertip. Two different types of tactile sensors which were based on different operating principles were used in order to show that the proposed framework was agnostic to the tactile sensing technology.

The normalised force values of the tactile sensors are measured and contact is established when the force exceeds the baseline threshold $f_{ts} \geq \tau_f$ where $\tau_f = 1.1$. The contact points P^t_{obs} expressed in the common world frame $\mathcal{W}$ are added to the tactile point cloud P^t after every action. An Azure Kinect DK©RGB-D camera is rigidly attached to the Panda Gripper with a custom designed flange which provides the vision point cloud. Hand-eye calibration is performed to find the transformation between the Panda Gripper and the camera frame and consequently transformed into the common world coordinate frame $\mathcal{W}$. All operations involving point clouds use the Point Cloud Library (PCL) (Rusu and Cousins, 2011), occupancy grid computations uses OctoMap library (Hornung et al., 2013), and the

Figure 5.10: a) Target unknown objects. The properties evaluated by human experts: T: Transparency/Specularity, C: Shape complexity, S: Symmetry, +: medium, ++: high. b) Objects used to clutter the workspace. c) Visuo-tactile point cloud of an exemplary object demonstrating the need for tactile exploration in regions of transparency where vision data is absent, (d) Visuo-tactile point cloud of a transparent object wherein visual data is completely missing and object is reconstructed and localised with tactile data.

overall setup uses a ROS-based framework. All robot experiments are run on a workstation using Ubuntu 18.04 with Intel©Xeon Gold 5222 CPU. The reconstruction network is implemented using the Tensorflow framework and training/ inference are performed on Nvidia Quadro RTX 4000 GPU. The maximum allowed speeds for the UR5 and Panda were 75 mm/s and 100 mm/s respectively for safety constraints.

Object List: In order to be easily reproducible, widely available daily objects are used for experimentation from the following categories: (a) bottle, (b) cup, (c) mug, (d) spray, (e) detergent and (f) wineglass. The objects from each category are shown in Fig. 5.10a. These objects are unknown and their models are reconstructed and pose estimation is performed. Furthermore, a set of other objects shown in Fig. 5.10b are used to clutter the workspace and the target object. Each scene is composed of one target object from any category and a subset of clutter objects placed around the target object in randomised dense clutter scenarios.

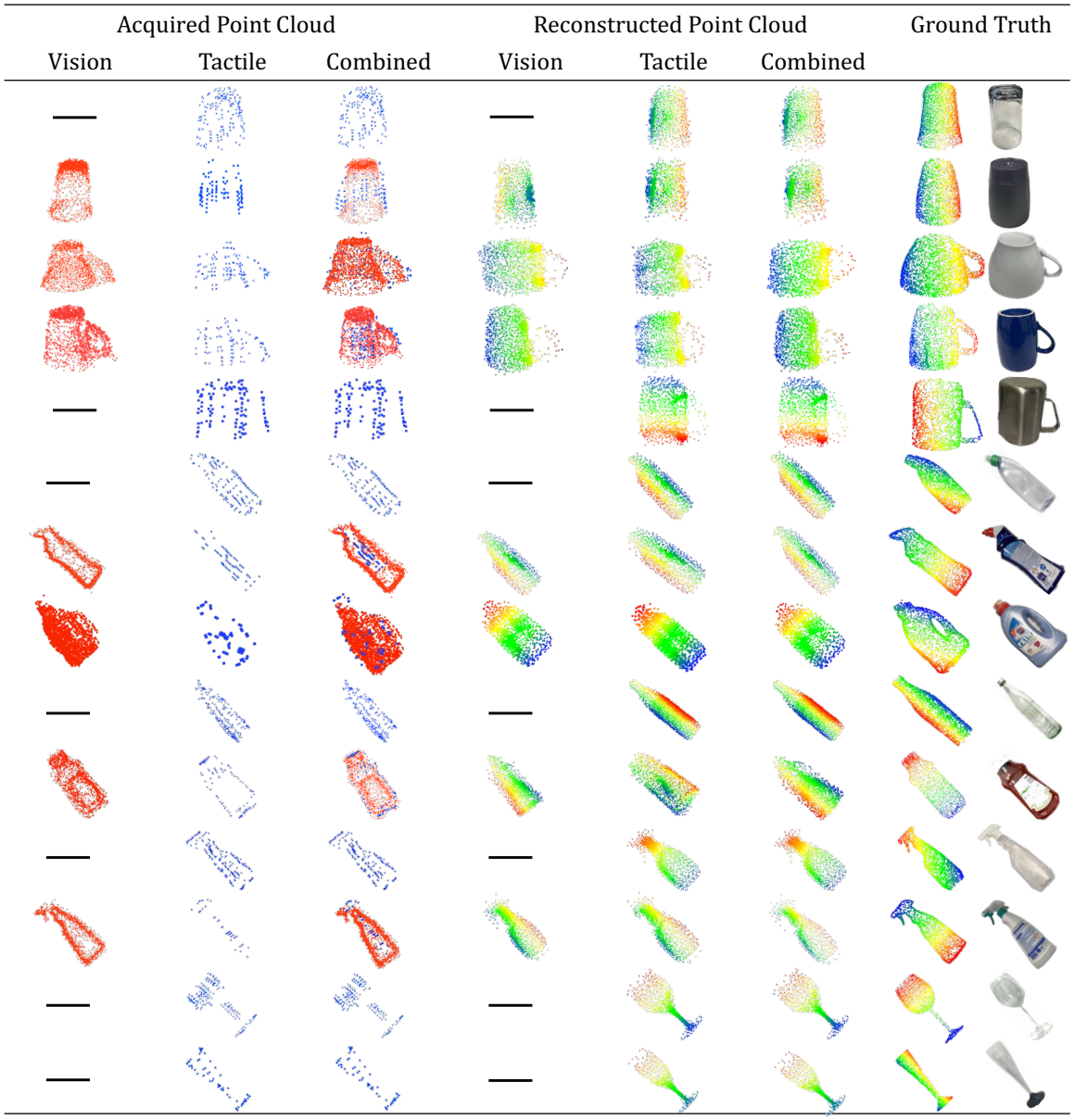


Figure 5.11: Visuo-tactile point clouds and the respective reconstructed point cloud using the proposed reconstruction network. The first and second column shows the measured point clouds from vision and tactile sensing respectively. The third column (combined) shows the union of the visuo-tactile point clouds after being aligned with each other. The reconstructed point clouds (columns 4-6) shows the output of the reconstruction network for vision, tactile and visuo-tactile input point clouds. The last column shows the ground truth point cloud sampled from the corresponding CAD mesh and the respective RGB images.

5.3.2 Active Visuo-Tactile based Target Object Reconstruction

As the target objects list contains both transparent and opaque objects, the framework automatically prefers tactile exploration for transparent objects and visuo-tactile explo-

ration for opaque objects using the joint criteria defined in Eq. (5.10). An exemplary case that demonstrates the benefit of the proposed method is shown in Fig. 5.10c. The ketchup bottle has parts of transparent and non-transparent regions. The vision point cloud shown in red captures the overall shape but contains missing points in the transparent region (highlighted in the green box). Due to the information gain method for object exploration, tactile acquisitions are performed only in the regions where the visual points are missing or there is uncertainty due to noisy data (around the edges).

For reconstruction evaluation, the Chamfer distance metric (CD) compared to the ground truth point clouds is used. The ground-truth point clouds shown in Fig. 5.12 are extracted using a specialized hand-held sensor device. The chamfer distance metric (CD) is defined as the sum of the average nearest-neighbour distance between one point cloud and the other and vice-versa. Mathematically, given the ground-truth point cloud $\mathcal{P}_{gt}$ and the reconstructed point cloud $\mathcal{P}_{recon}$, the CD is defined as:

$$CD(\mathcal{P}_{gt}, \mathcal{P}_{recon}) = \frac{1}{|\mathcal{P}_{gt}|} \sum_{p_1 \in \mathcal{P}_{gt}} \min_{p_2 \in \mathcal{P}_{recon}} ||p_1 - p_2||_2 + \frac{1}{|\mathcal{P}_{recon}|} \sum_{p_2 \in \mathcal{P}_{recon}} \min_{p_1 \in \mathcal{P}_{gt}} ||p_2 - p_1||_2 \tag{5.23}$$

where $|\bullet|$ refers to the cardinality (number of points) of the point cloud and $||\bullet||_2$ refers to the L2 norm. Lower CD value denotes higher reconstruction precision. For the ideal case where the reconstructed point cloud exactly matches the ground truth, $CD \approx 0.0$ m. Qualitatively, through empirical analysis, it was found that $CD < 0.01$ m denotes accurate reconstruction, 0.01 m $<$ CD $<$ 0.1 m denotes good reconstruction while $CD > 0.2$ m implies poor reconstruction of the point cloud. The qualitative results for the reconstruction with vision and tactile data with the proposed reconstruction network are shown in Fig. 5.11. The acquired point clouds with visual, tactile sensing and combined visuo-tactile point clouds (where each point cloud was aligned with each other and aggregated) were fed into the reconstruction network and the reconstructed point clouds were extracted as output shown in Fig. 5.11. For all transparent objects, visual point clouds were not captured due to sensitivity of depth cameras with transparent objects. Hence, only tactile point clouds were used for reconstruction. For opaque objects, the tactile point clouds were captured in regions of uncertainty of visual data. It can be seen that the proposed reconstruction network was capable of reconstructing from sparse tactile and dense visuo-tactile inputs in an accurate manner compared to the ground-truth point clouds which were sampled from the CAD mesh of the objects. The proposed reconstruction technique method, with the help of the learned model over the category-level synthetic objects, is able to reconstruct the ob-

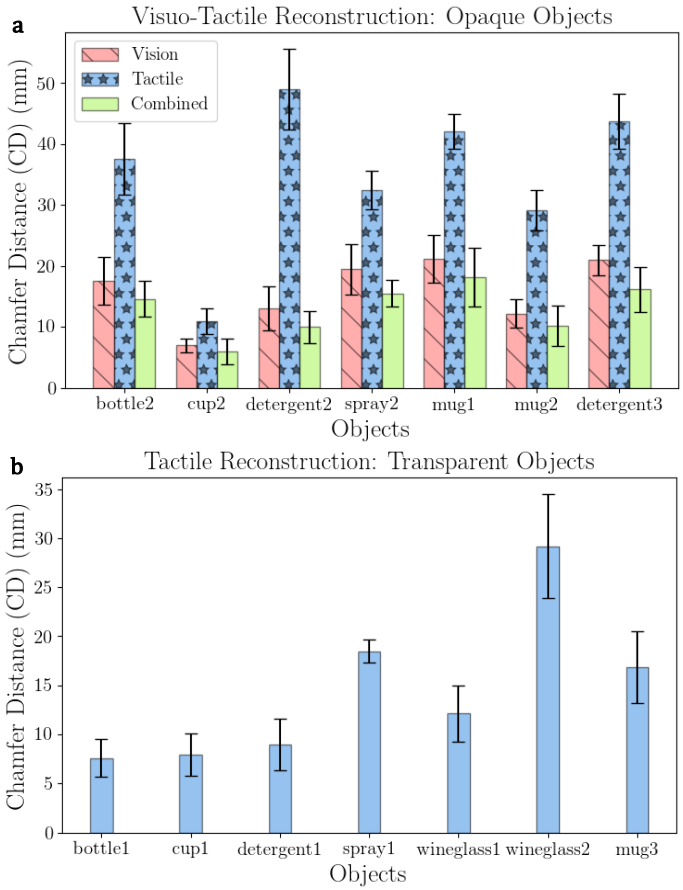


Figure 5.12: Quantitative reconstruction results showing the Chamfer distance (CD) metric of the reconstructed point cloud compared with the ground-truth point cloud for (a) opaque objects and (b) transparent objects. The bar graph represents the average values and the error bars represent the standard deviation.

ject even with sparse input point clouds. Five repeated experimental trials were conducted for each target object and with each exploration strategy: active, random, and uniform, resulting in 210 total trails (14 objects, 3 strategies, 5 repetitions). It can be seen that for opaque objects, the shared visual and tactile data result in a higher accuracy of reconstruction ($CD < 2$ cm) as seen in Fig. 5.12a. The visual point clouds for opaque objects capture all the sides of the objects due to the active visual exploration. On average, combining with tactile data improves visual reconstruction accuracy by 17%. For opaque objects, the tactile reconstruction accuracy is relatively worse due to the fact that incomplete tactile point clouds are collected as the robot only explores regions unseen by the camera.

For transparent objects, even a sparse input point cloud provides acceptable reconstruction accuracies as seen in Fig. 5.12b. In this case, since no visual point cloud is available, the robot explores the object with only tactile sensing in an information-gain seeking

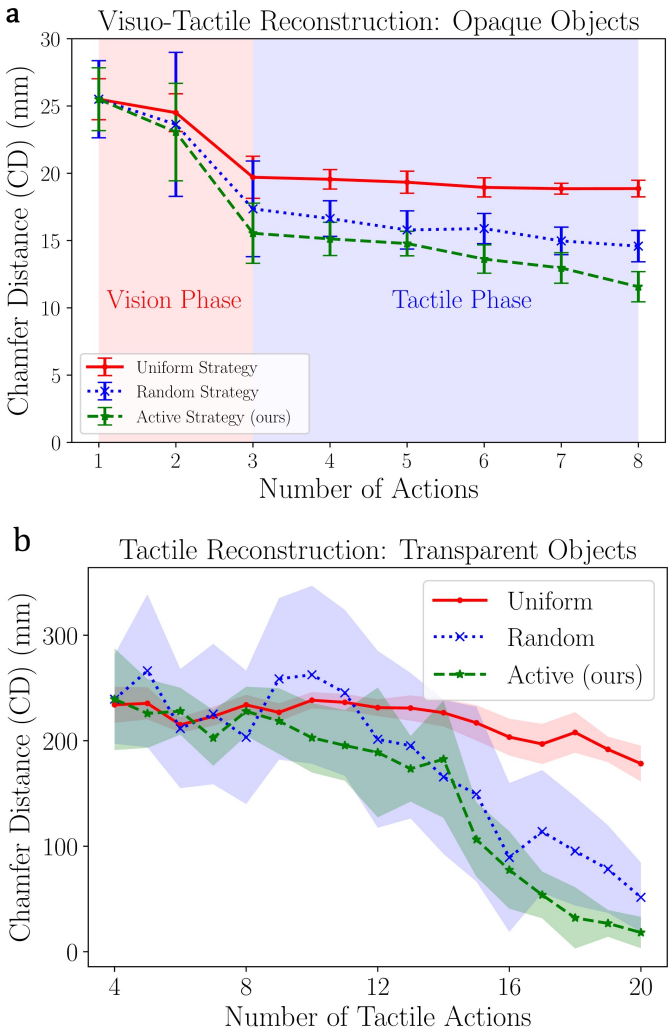


Figure 5.13: (a) Active visuo-tactile reconstruction accuracy for opaque objects and (b) active tactile-only reconstruction accuracy for transparent objects compared with random and uniform strategies. The error bars in (a) and shaded regions in (b) represent the standard deviation.

strategy. All objects are accurately reconstructed ($CD < 2$ cm) by the network except the `wineglass2`. The reason for the lower accuracy for `wineglass2` is due to its peculiar shape, and it is out of distribution to the training dataset.

Furthermore, a comparison study of the proposed active exploration strategy with baseline random and uniform strategies was performed for both vision and tactile modality. The baseline strategies for tactile exploration are defined as follows: the bounding box on the target object is discretised into a 3D grid with each grid cell of size 3 cm$\times$3 cm which corresponds to the size of the sensor patch. The robot is moved to touch the grid cell closest to its base frame and sequentially touches each cell in a uniform manner. In contrast,

the random strategy involves choosing the next possible grid cell in a randomised manner. In a similar manner, random and uniform exploration strategies are defined for the vision modality: viewpoints on the hemisphere sphere are sampled uniformly in the same way as described in Sec. 5.2.3 and the robot starts from one extreme possible position and sequentially moves to the next viewpoint in a uniform manner. The random strategy chooses one among the possible sampled viewpoints at random. As the objective is to compare the sample efficiency of the actions and to have an unbiased comparison with each strategy, the number of tactile probing actions were limited to 20 actions for transparent objects and to 3 visual actions and 5 tactile actions for the remaining opaque objects. The results for reconstruction of opaque objects with visuo-tactile sensing is shown in Fig. 5.13a and transparent objects with tactile sensing in Fig. 5.13b. As the exploration is performed for the objective of object reconstruction, CD is used as a metric for comparison. For both transparent and opaque objects, increasing the number of exploratory actions reduces the CD value. For transparent objects that rely upon only the sense of touch, active exploration converges to an accuracy of $CD \approx 2$ cm within 20 touches. Random exploration converges to $CD \approx 5$ cm in 20 touches, but with higher variance due to the stochasticity of exploratory actions. Uniform exploration has the least accuracy due to the fixed nature of exploration, which often collects redundant data, leading to long data collection times. In contrast for opaque objects, random and active strategies perform similarly on average ($CD \approx 1.5$ cm) and subsequently active tactile strategy slightly improves the reconstruction accuracy ($CD \approx 1.1$cm). Negligible reduction in CD value is reported with random and uniform tactile actions following visual perception for opaque objects.

5.3.3 Category-level Visuo-Tactile based Pose Estimation

In order to benchmark the stochastic TIQF (S-TIQF) and previously presented TIQF methods, firstly instance-level pose estimation was performed where the object model point cloud is obtained from the ground truth mesh on the Stanford Scanning Repository benchmark. The following state-of-the-art methods are used for comparison: Iterative Closest Point (ICP) (Besl and McKay, 1992), Sparse iterative closest point (S-ICP) (Bouaziz et al., 2013), Random sample consensus (RANSAC) (Fischler and Bolles, 1981), Truncated least squares Estimation And SEmidefinite Relaxation (TEASER++) (Yang et al., 2020) and PREDATOR (Huang et al., 2021a). Both local registration methods such as ICP and S-ICP, global optimization methods such as RANSAC and TEASER++ and learning methods such as PREDATOR are compared against. These popular baselines are chosen as they are often used in the literature for the point cloud registration task. Furthermore, some of these baselines such as ICP and RANSAC are also used to perform the final registration

task with learning-based methods where the features are learnt using a neural network. The learning-based method termed PREDATOR (Huang et al., 2021a) learns to predict the registration of point clouds with low overlap between each other as is the case with visuo-tactile point clouds. The pretrained model of PREDATOR (Huang et al., 2021a) was used and hyper-parameters were set as suggested in the paper with the exception of `first_subsampling_dl` = 0.01, `dgcnn_k` = 5 for sparse point clouds. Secondly, to demonstrate the flexibility of the proposed method, the PhoCal dataset (Wang et al., 2022a) was used to perform a feasibility study for category-level pose estimation using the NOCS-based (Wang et al., 2019) framework. For real robot experiments, the reconstructed point clouds of the objects from the reconstruction network were used as the object point cloud and the acquired vision and/ or tactile point cloud as the scene point cloud for category-level pose estimation.

Benchmark Experiments

Stanford Scanning Repository Benchmark: In order to benchmark the presented methods against the state-of-the-art, a standard point cloud registration benchmark from the Stanford Scanning repository (Levoy et al., 2005) was used. Six CAD models from the dataset namely bunny, dragon, happy Buddha, Lucy, statue and armadillo (Levoy et al., 2005) (cf. Fig. 4.3a for the objects from the Stanford Scanning Repository). In order to

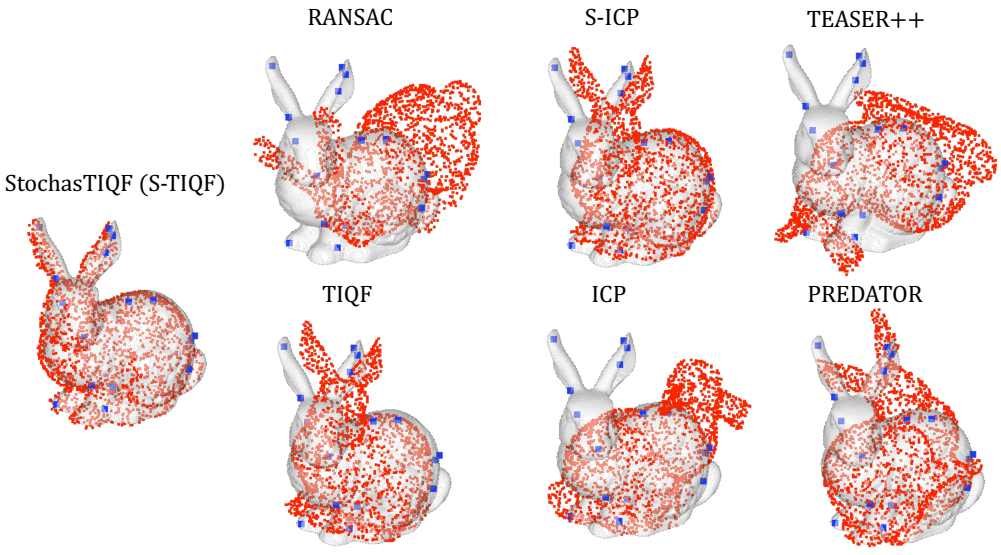


Figure 5.14: Qualitative results on the Stanford Bunny Dataset: The grey mesh represents the model at ground truth for reference, the blue sparse point cloud represents the scene point cloud and the red dense point cloud represents the transformed model point cloud after performing point cloud registration.

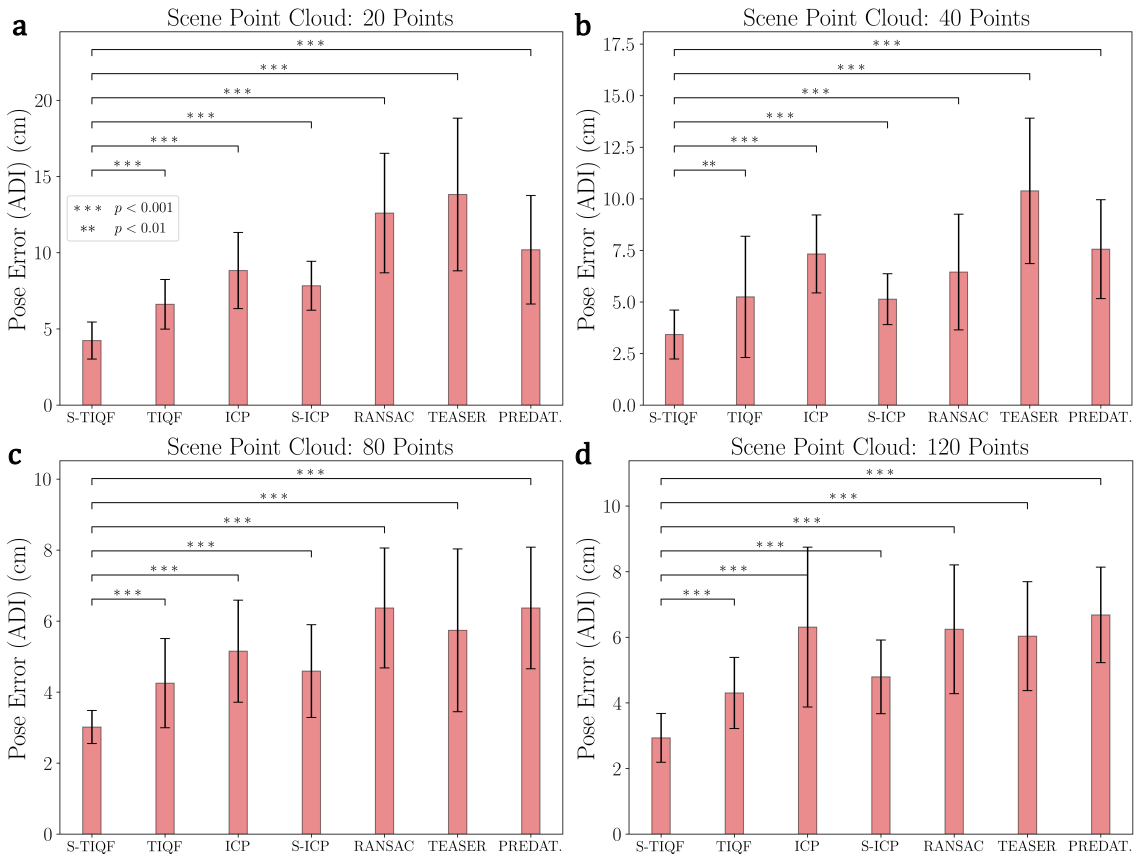


Figure 5.15: Pose error calculated as ADI error for models from the Stanford Scanning Repository. The object point cloud consisting of 1024 points is sampled from the models while the scene point cloud is randomly sampled from the model and consists of (a) 20, (b) 40, (c) 80 and (d) 120 points respectively. p values calculated by Welch's t-test shown as $*$. The bar plot represents the average and the error bars represents the standard deviation.

have an unbiased comparison of pose estimation, the model point cloud are derived from the CAD mesh in the dataset. This is done because errors in shape reconstruction can propagate and influence pose estimation. Each model point cloud is sampled uniformly from the CAD mesh to have 1024 points. The scene point cloud is sampled randomly from the CAD mesh and the point numbers are set to 20, 40, 80 and 120 points. The varying degree of sparsity can test the robustness of the proposed approach against state-of-the-art methods. The model and scene point clouds are normalized and scaled to lie within a $[-1,1]^3$m cube. In order to evaluate the sensitivity of the proposed method against local optima, the initial pose for the model point cloud is randomly chosen from a position range of $[-5.0,5.0]$m and rotation from $[-180°,180°]$ for each experimental trial. The correspondence estimation for ICP and S-ICP is based on nearest neighbourhood search whereas RANSAC and TEASER++ are based of Fast Point Feature Histograms (FPFH) descriptors (Rusu et al.,

2009). For each selected model from the Stanford scanning repository, the experiment is repeated 5 times with the initial pose randomly varied for each trial. The errors are measured as using the Average Distance of model points with Indistinguishable views metric (ADI) which is insensitive to object symmetries (Hinterstoisser et al., 2013). The ADI metric is measured as:

$$\texttt{err}_{adi} = \frac{1}{|\mathcal{O}|} \sum_{\mathbf{p}_1 \in \mathcal{O}} \min_{\mathbf{p}_2 \in \mathcal{O}} ||(\mathbf{R}_{gt}\mathbf{p}_1 + \mathbf{t}_{gt}) - (\mathbf{R}_{est}\mathbf{p}_2 + \mathbf{t}_{est})|| \tag{5.24}$$

where $(\mathbf{R}_{gt}, \mathbf{t}_{gt})$ and $(\mathbf{R}_{est}, \mathbf{t}_{est})$ refers to ground-truth and estimated rotation and translation respectively, $\mathcal{O}$ refers to the object model point cloud and the points $p_1 \in \mathcal{O}$ and $p_2 \in \mathcal{O}$ belong to the object point cloud and denote the closest corresponding points when $\mathcal{O}$ is transformed by $\{\mathbf{R}_{gt}, \mathbf{t}_{gt}\}$ and $\{\mathbf{R}_{est}, \mathbf{t}_{est}\}$ respectively.

Qualitative results with the Stanford Bunny model are shown in Fig. 5.14. Quantitative results evaluated with all the models selected from the Stanford scanning repository are provided in Fig. 5.15. It can be seen that for all levels of point sparsity (20-120 points), the S-TIQF outperforms baselines ($p < 0.001$ for Welch's t-test in all cases except for scene cloud with 40 points with TIQF where $p < 0.01$). Interestingly, S-TIQF also outperforms TIQF method and this is due to the stochastic initial alignment used in S-TIQF. For instance, in the case of the scene point cloud with 20 points, S-TIQF outperforms the closest baseline S-ICP by 45% on average and 38% for scene point cloud with 120 points. The results corroborate the known weaknesses of correspondence-based techniques such as RANSAC and TEASER++ as they rely upon features for estimation correspondences. These correspondences remain fixed throughout the pose estimation process. Due to point sparsity, feature extraction methods such as FPFH fail to generate valid correspondences. Similarly, the S-TIQF and TIQF methods outperforms the learning-based method PREDATOR by more than 50% on average for all levels of point sparsity (20-120). The point sparsity and absence of neighborhood points is challenging for the graph neural network in PREDATOR to extract good features for the overlapping regions. Furthermore, simultaneous pose and correspondence methods such as ICP and TIQF perform relatively well on sparse data but rely on good initialization. The S-TIQF approach removes the need for good initialization through the stochastic search for initial alignment.

Ablation Study: The effect of the stochastic alignment approach based on Simulated Annealing (SA) is compared with other local (or fine) registration methods such as ICP and S-ICP which is term as SA+ ICP and SA+ S-ICP. Similar to the evaluation procedure above, the bunny dataset from the Stanford Scanning Repository is used and an initial pose is randomly sampled from [-5.0,5.0]m and $[-180^\circ, 180^\circ]$ and repeated the experiment 20

times. While in the evaluation above, the randomly sampled initial pose is provided as input to the proposed approach and all baseline methods, in this case, the initial pose was provided to the stochastic alignment method and the baseline methods (ICP and S-ICP) are suitably modified such that they take as input the pose output from the SA module. The results are presented in Tab. 5.1. The SA approach improved the pose estimation for both ICP and S-ICP methods for all levels of point sparsity: for instance, pose estimation with scene point cloud with 20 points improved by $\sim 10\%$ whereas for 120 points, it improved by over 25%. The stochastic alignment method also improves with increasing number of points. However, the S-TIQF approach still outperforms the modified SA+ICP and SA+S-ICP approaches by atleast 20% in terms of ADI error.

PhoCal Dataset Benchmark: A feasibility study with the PhoCal dataset (Wang et al., 2022a) is conducted to demonstrate category-level pose estimation with the S-TIQF method. In contrast to the proposed reconstruction network which is trained on point clouds of synthetic objects belonging to the same category as the real-world objects, the Normalized Object Coordinate Space (NOCS) based framework (Wang et al., 2019) can also be used to generate the point cloud of the objects. Given RGB inputs, the NOCS network learns a NOCS map which is a shared canonical space of objects within a category. The NOCS map can be combined with the depth map to lift from 2D image to 3D point cloud space. This is used as the object point cloud and the point cloud from the depth map is considered as the scene point cloud for point cloud registration. Furthermore, point cloud registration methods such as Umeyama algorithm (Umeyama, 1991) are used to perform pose estimation with the NOCS-based framework (Wang et al., 2019). The PhoCal dataset (Wang et al., 2022a) contains the RGB, depth and learnt NOCS maps of real world objects belonging to different categories particularly for photometrically challenging objects. In order to perform accurate 6D pose annotation, the authors of the PhoCal dataset (Wang et al., 2022a) used a tool-tip on a robotic manipulator to manually touch

Table 5.1: Ablation study for the effect of the stochastic alignment with Simulated Annealing (SA) method on ICP and S-ICP. The values presented are mean error and standard deviation.

Scene Pt. Size	**S-TIQF ADI (cm)**	**SA + ICP ADI (cm)**	**SA + S-ICP ADI (cm)**
20 points	4.36 ± 1.32	7.65 ± 2.23	6.98 ± 1.54
40 points	3.35 ± 1.21	6.61 ± 2.08	5.11 ± 1.64
80 points	3.05 ± 0.85	5.41 ± 1.53	4.21 ± 1.35
120 points	2.85 ± 0.79	4.28 ± 1.67	3.55 ± 1.58

the object at various locations that are sparsely distributed on the object. These touch points are used as the tactile point cloud in this work. The learnt NOCS maps are used to generate the object model point cloud and the depth map provides the visual point cloud. S-TIQF is compared with method introduced in (Wang et al., 2019) for pose estimation. Fig. 5.16 shows an example from the PhoCal dataset demonstrating the rendered NOCS map (Fig. 5.16b) and the reconstructed models (Fig. 5.16d). The reconstructed point clouds are partial (see bottle and fork in Fig. 5.16d) as only the visible portions of the scene are used to generate the NOCS map and provides further challenges for pose estimation. Fig. 5.17 shows the comparison results of S-TIQF method against the Umeyama approach (Umeyama, 1991) used in (Wang et al., 2019). It is shown that the proposed approach outperforms the baseline method for tactile point clouds by approximately 35% median ADI error and about 20% ADI error when applied to dense visual point clouds ($p < 0.001$). The scale estimation approach that is used for visual and tactile sensing based point clouds is shown in Fig. 5.17 on the secondary Y-axis. The scale error is calculated as:

$$\texttt{err}_{scale} = \sqrt{(s_x^{gt} - s_x^{est})^2 + (s_y^{gt} - s_y^{est})^2 + (s_z^{gt} - s_z^{est})^2} \tag{5.25}$$

wherein, $\mathbf{S}^{est} = \{s_x^{est}, s_y^{est}, s_z^{est}\}$ represents the estimated scale and the $\mathbf{S}^{gt} = \{s_x^{gt}, s_y^{gt}, s_z^{gt}\}$ denotes the ground-truth value. Lower error denotes better estimation. On average, the scale error is 43% lower for tactile-based perception compared to the visual perception. As the dataset consists of transparent and specular objects, the visual sensing method provides incomplete point clouds with a large number of missing or erroneous points whereas the tactile sensing approach provides sparse but approximately uniformly distributed point clouds which can be seen from Fig. 5.16c. Furthermore, tactile sensing is insensitive to the transparency or specularity properties of the objects and provides high fidelity point measurement.

Robotic Experiments

In order to validate S-TIQF in real world settings, extensive experiments were carried out using the robotic setup shown in Fig. 5.1 and everyday objects shown in Fig. 5.10. Similar to the benchmark experiments, the S-TIQF and TIQF methods are compared against the same baseline methods. The model point cloud is derived from the reconstructed point cloud from the reconstruction network. The scene point cloud comprises of vision and/or tactile data. For each target object, the experiment is repeated 5 times by randomising the cluttered scene for each iteration. Similar to the previous experiments, the initial pose is sampled randomly from [-5.0,5.0] m and $[-180^\circ, 180^\circ]$ for each trial and the same ini-

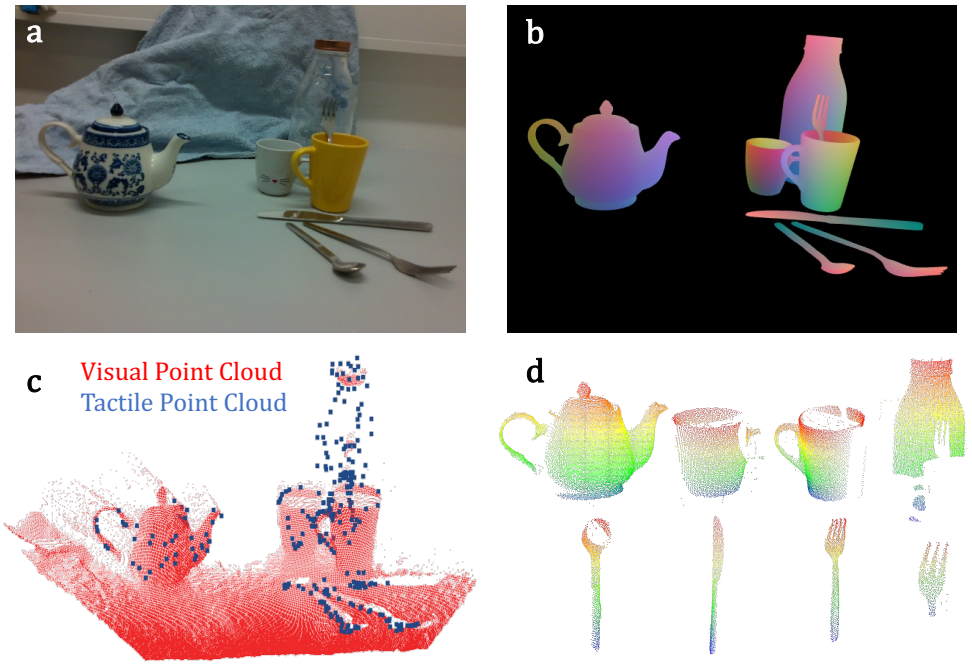


Figure 5.16: Qualitative results using the PhoCal dataset: (a) RGB input, (b) rendered NOCS map, (c) Visual and tactile point cloud, (d) reconstructed model point clouds from NOCS maps in (b).

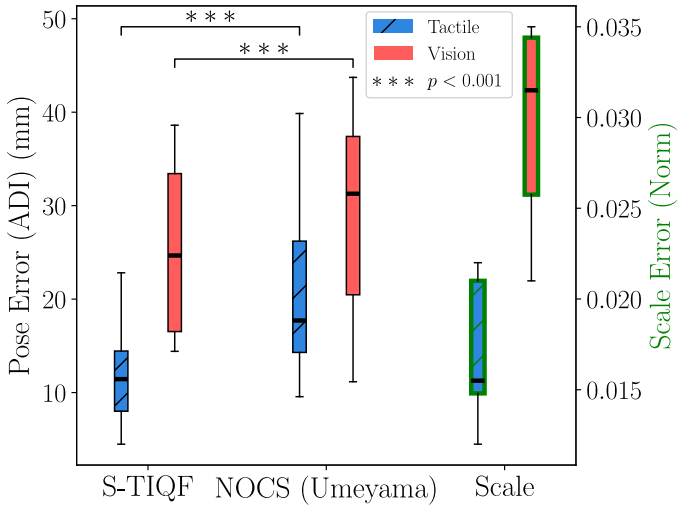


Figure 5.17: Comparison of the proposed method against the NOCS (Umeyama method) (Wang et al., 2019) performed as a feasibility study with the PhoCal dataset. p values calculated by Welch's t-test shown as ∗.

tial pose is provided to all comparison methods. The quantitative results for transparent objects are shown in Fig. 5.18a and opaque objects in Fig. 5.18b. The results with transparent objects are similar to the benchmark experiments due to the nature of sparse tactile

point clouds and S-TIQF outperforms the baseline approaches. For instance, S-TIQF outperforms the next best baseline method S-ICP by nearly 40% on average for sparse tactile point clouds ($p < 0.001$). In comparison, it can be seen from Fig. 5.18b that for dense visual point clouds, nearly all the methods perform equally well and the S-TIQF method compares favourably with the state-of-the-art ($p < 0.001$). The proposed method achieves an average ADI error of 2.1 cm whereas TEASER++ achieves an error of 2.5 cm for dense visuo-tactile point clouds ($p = 0.04034$). The learning-based approach PREDATOR also performs on-par with other baselines for the dense visuo-tactile point clouds with average ADI error of 3.1 cm. However, the performance of PREDATOR with sparse tactile point clouds for transparent objects are much worse, with the average accuracy of S-TIQF nearly 65% better than PREDATOR. The PREDATOR method assumes sufficiently dense local point features even if there is minimal overlap for the overlap attention module which is not the case for sparse tactile point clouds and results in lower performance. In fact, it can be seen that the combined visuo-tactile point clouds results in better accuracy than visual or tactile point clouds alone, demonstrating the importance of shared perception. For instance, the accuracy improves by $\sim 35\%$ for S-TIQF using the combined visuo-tactile point cloud instead of either vision or tactile point clouds alone. In this case, it should be noted that the higher levels of inaccuracies with tactile data are a result of the active object exploration strategy wherein tactile data are only collected in locations of higher uncertainty and inaccessible locations to visual data. Hence, it can be concluded that the S-TIQF method provides highly accurate pose estimation for both sparse tactile and dense visual data. The object-wise pose estimation results with S-TIQF is shown in Fig. 5.19 and detailed in the Discussion section (Sec. 5.4).

5.3.4 Visuo-Tactile Hand-Eye Calibration

This section provides comparative studies performed for hand-eye calibration of the proposed *in-situ* approach and standard methods using calibration grid with the algorithm originally presented by Tsai and Lenz (1989). For the calibration grid method, the grid was fixed at a suitable distance from the camera such that it is clearly within the field of view of the camera. Ten different viewpoints were chosen manually ensuring that the different end-effector rotations were incorporated. The experiment was repeated five times. The *in-situ* visuo-tactile calibration approach does not require a specialized grid. Any object in the workspace of the robot can be used as long as an accurate point cloud corresponding to the object is available. The object must be immobilised and multiple visual pointclouds are captured from different viewpoints. With an incorrect hand-eye calibration, the point clouds from different views would not overlap accurately and result in the scenario shown

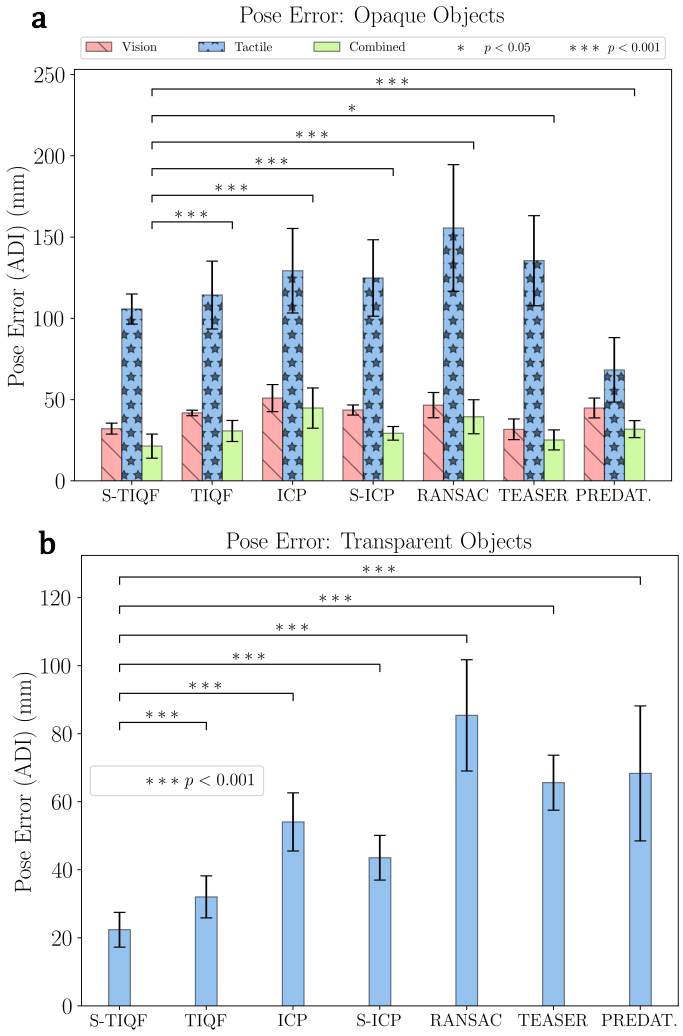


Figure 5.18: Average pose error for real-world objects for (a) opaque objects with visuo-tactile perception and (b) transparent objects with tactile perception. p values calculated by Welch's t-test shown as ∗.

in Fig. 5.20a. Furthermore, when tactile data are collected from the same object, resulting in the tactile point cloud, an incorrect hand-eye calibration can be described in the scenario shown in Fig. 5.20b. The tactile sensors are rigidly attached to the end-effector and the robot kinematics are accurate enough to provide a grounding of the object pose. Using the calibration grid method, an acceptable accuracy can be achieved, but residual errors would still be present. The qualitative results are shown in Fig. 5.20c. Using the in-situ approach with the S-TIQF method, a highly accurate solution can be obtained (Fig. 5.20d). Quantitative results are shown in Fig. 5.20e. The proposed approach achieves < 1 cm error in position and $< 5°$ error in rotation.

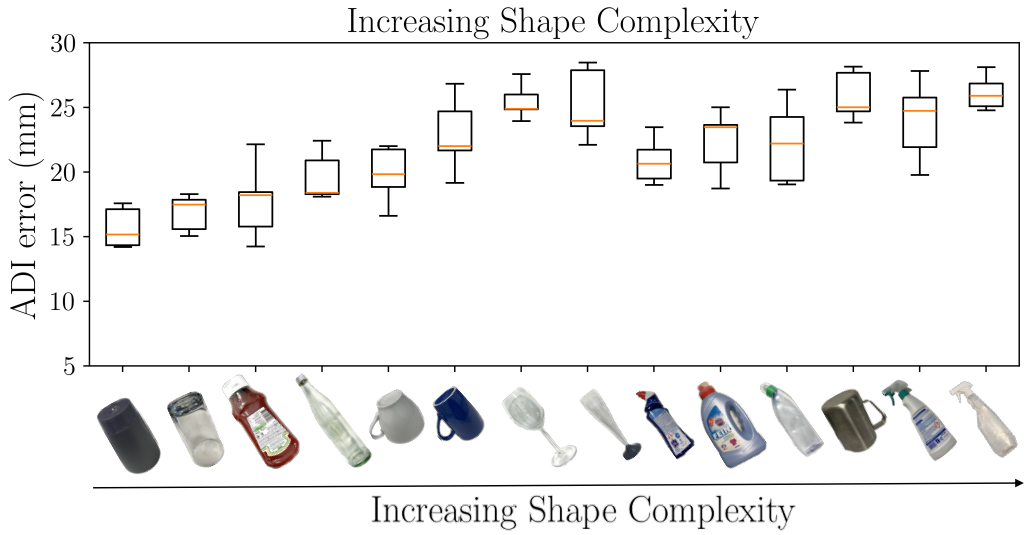


Figure 5.19: Object-wise pose estimation results with S-TIQF

Table 5.2: Numerical results from kinematic calibration benchmark showing the median error and median absolute deviation

	UR5	**Panda**
Calibration Error (mm)	0.303 ± 1.82	0.432 ± 3.342

Robot Kinematic Accuracy Benchmark for Calibration

The in-situ visuo-tactile based hand-eye calibration method depends on the accuracy of the kinematic calibration (especially for the robot with tactile sensing). Although this assumption is commonly used in the case of hand-eye calibration (Sun and Hollerbach, 2008), the kinematic accuracy of both robots were benchmarked using an external sensor system to evaluate the effect on hand-eye calibration. The OptiTrack motion capture system (NaturalPoint, Inc.) was used to track a specially designed coordinate frame with embedded markers that is attached to the gripper fingers, as shown in Fig. 5.21a. The motion capture system has an average accuracy of 0.1 mm. The UR5 robot, which is sensorized with tactile sensors, has a pose repeatability of 0.1 mm and is kinematically calibrated by the manufacturer[1]. The Franka Emika robot which is sensorised with the camera also has a pose repeatability of 0.1 mm obtained from the datasheet of the manufacturer. Both robot's end effector were moved in arbitrary trajectories covering all 6 DoF by the human user with manual hand guiding and the pose of the end-effector was extracted from the kinematic model and using the motion capture system, respectively. The initial static offset between

[1] `https://www.universal-robots.com/articles/ur/robot-care-maintenance/kinematic-robot-calibration/`

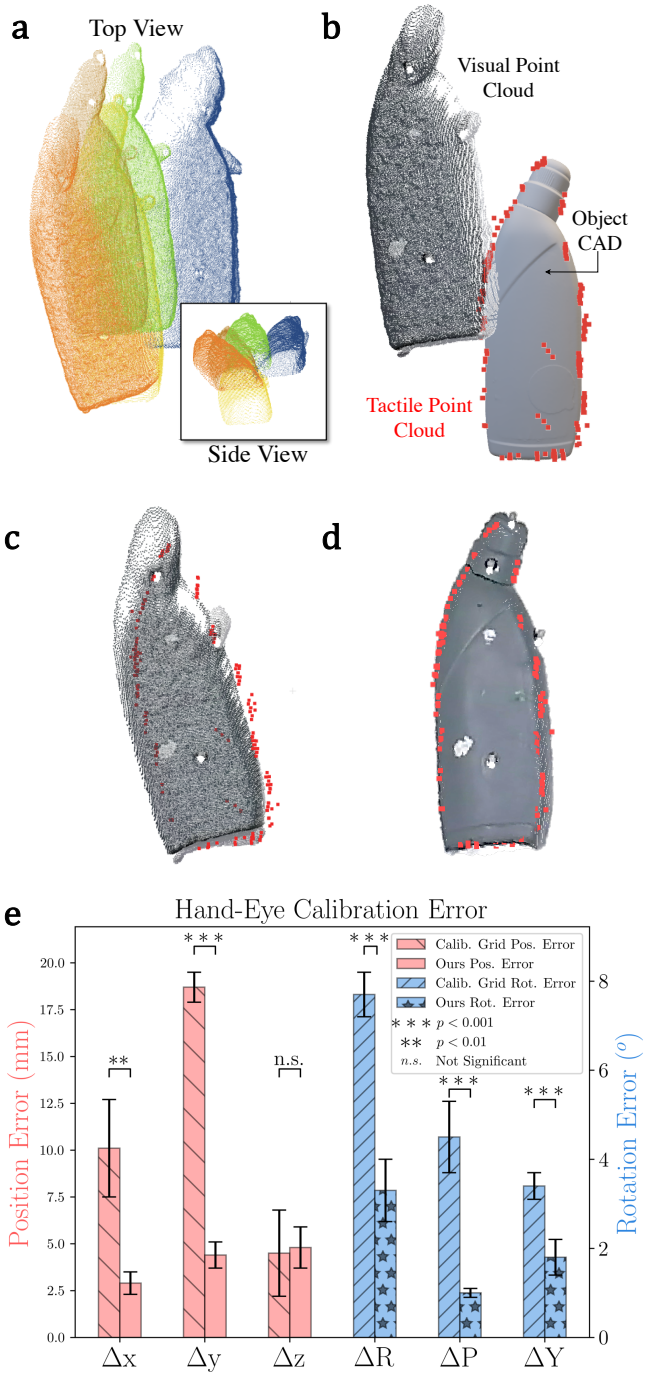


Figure 5.20: Qualitative results of hand-eye calibration: Effects of incorrect calibration when point clouds are acquired from different viewpoints (a) and (b). The different colours for the point clouds in (a) highlights the effect of incorrect calibration when overlapped with each other. The accuracy of calibration using grid-based method (c) and the proposed method (d). Quantitative analysis of the error in hand-eye calibration (position and rotation) (e). p values calculated by Welch's t-test shown as ∗.

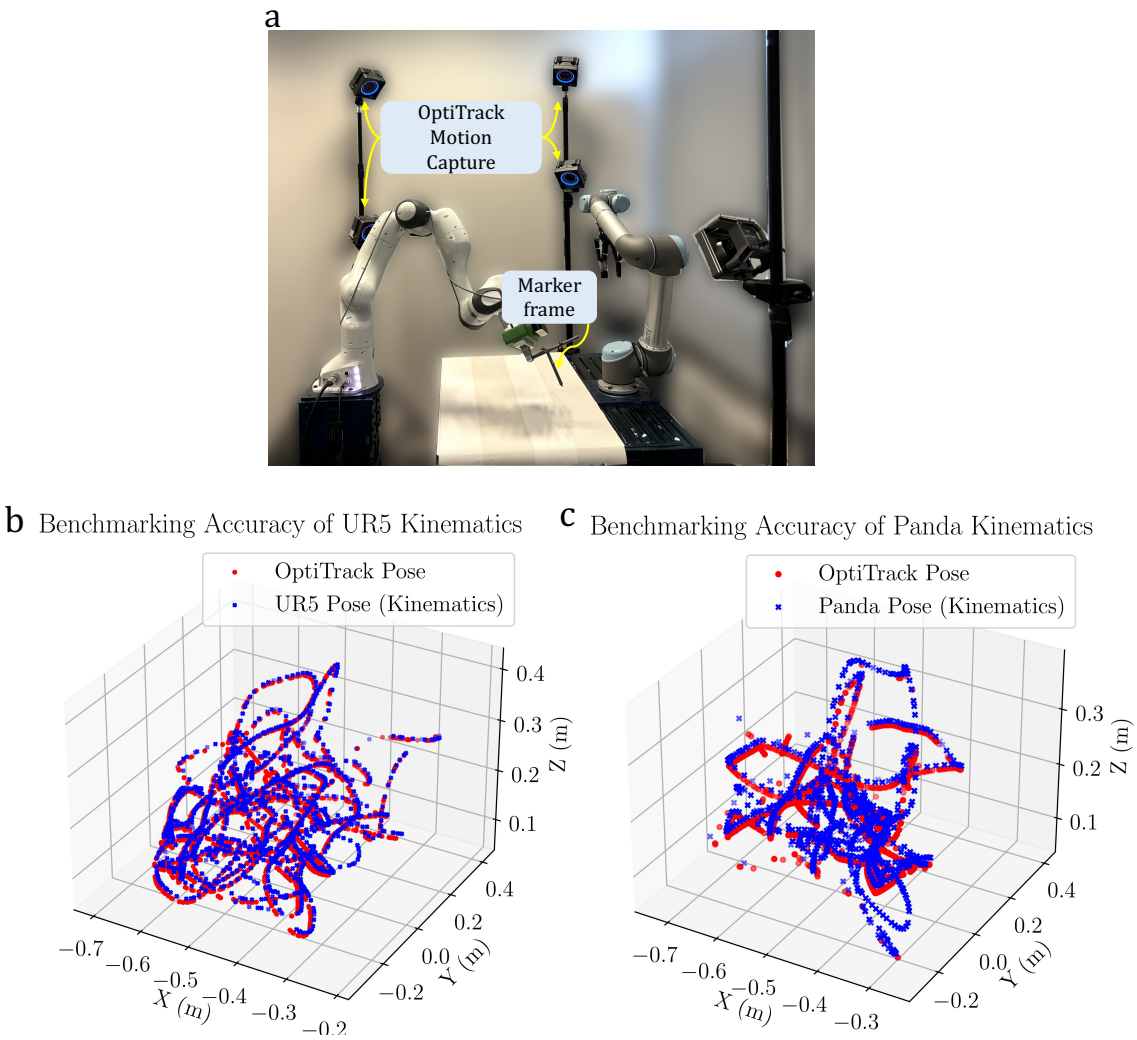


Figure 5.21: (a) Benchmarking the robot kinematics with high-precision marker-based motion capture system (OptiTrack) for (b) UR5 robot and (c) Franka Panda Robot. The robot poses are shown in blue and the pose calculated by the motion capture system in red.

the pose measurements is calculated and compensated. The accuracy between these poses in Fig. 5.21b,c are compared where the plots of the end-effector trajectory are shown with the kinematic pose calculation in blue and the motion capture calculated pose in red. The numerical results are shown in Tab. 5.2. There are intrinsic uncertainties inherent in the comparative analysis: sporadically, the human operator might occlude the markers from the field of view of certain OptiTrack cameras during manual guidance, despite the strategic deployment of six OptiTrack cameras surrounding the workspace. Hence, the median error and median absolute deviation were measured to disregard spurious outlier points. It can be seen that the kinematic accuracy of the UR5 robot is 0.303 ± 1.82 mm, which is crucial for the calculation of the tactile point clouds. This accuracy for the tactile mea-

surements is within the tolerance bounds. The kinematic discrepancies observed in the Panda robot are more pronounced ($\pm$4 mm median absolute deviation); however, their impact on the hand-eye calibration process remains minimal, as the tactile point cloud serves as the reference for the registration of the corresponding visual point cloud. The presented method works identically for the case where the camera is kept static and visual point cloud of the object is registered to the tactile point cloud.

5.4 Discussion

5.4.1 Individual Sub-System Evaluation:

In this chapter, a novel interactive shared visuo-tactile perception approach for unknown target object reconstruction and robust pose estimation has been presented. To retrieve the target object, the robots coordinated together to declutter the scene. The target objects of varying shape complexity and transparency were used to extensively evaluate the reconstruction and pose estimation pipeline. The proposed approach was able to accurately reconstruct both transparent and opaque novel objects efficiently in an active information-gain seeking manner. From Fig. 5.13a for transparent objects, it can be inferred that the uniform strategy required a large number of tactile actions for accurate reconstruction of the objects leading to increased data collection time. The random exploration strategy had a high standard deviation ($CD \approx 3$ cm after 20 actions) that was due to the stochastic nature of the exploration. The proposed active strategy had lower variance and higher accuracy ($CD < 2$ cm) within 20 actions, outperforming both random and uniform strategies. For vision based object reconstruction, due to the workspace limitations, wide field of view of the camera, and limited size of the objects, on average 3 viewpoints were sufficient to completely explore the objects. However, as seen from Fig. 5.13b, uniform strategy was less accurate than random and active strategy for visual reconstruction. Furthermore, subsequent tactile actions after visual perception improved the reconstruction accuracy by 17% with the proposed active strategy, whereas the improvement was marginal with random and uniform strategies. The acquired tactile data with random and uniform strategies were redundant with the visual point cloud data whereas with active strategy, the regions unexplored by the visual modality was explored with the tactile modality. Shared visuo-tactile perception proved more advantageous than sole reliance on mono-modal visual or tactile perception, and the proficient sharing of perceptual attributes between modalities demonstrates efficacy across various object types, including both transparent and opaque entities. Furthermore, active perception was required for effective shared perception to avoid redundant data collection and overlap between sensing data.

Similarly, for category-level pose estimation with the reconstructed point clouds of real-world objects, the S-TIQF method outperformed all the baseline strategies for tactile-based pose estimation due to sparsity of tactile point clouds with an average ADI error around 2 cm as seen from Fig. 5.18a ($p < 0.001$). For opaque objects, S-TIQF compared favourably to state-of-the-art methods for dense visual and visuo-tactile point clouds ($p < 0.05$). Comparison was performed with geometry-based point cloud registration methods such as ICP (Besl and McKay, 1992), S-ICP (Bouaziz et al., 2013), RANSAC (Fischler and Bolles, 1981) and TEASER++ (Yang et al., 2020) which does not have any neural network learning component as well as with PREDATOR (Huang et al., 2021a) which was a learning-based registration method. Furthermore, some of these popular baselines such as ICP and RANSAC were also used as a "backend" to perform the final registration task with learning-based methods where the features were learnt using a neural network. In fact, even the PREDATOR method (Huang et al., 2021a) learnt the feature points where there was maximal overlap between the point clouds and RANSAC was used to extract the final pose estimate using the correspondences. The S-TIQF approach allowed incorporating sparse as well as dense measurements for pose estimation. S-TIQF also outperformed TIQF by 35% for tactile and visuo-tactile-based pose estimation. The proposed stochastic initialization strategy proves effective for escaping local minima. It can be seen from Fig. 5.19 showing the object-wise pose estimation results that increasing shape complexity resulted in marginal reduction in pose accuracy. Transparent objects showed higher average errors compared to opaque objects as they relied solely upon tactile perception resulting in sparse data. However, the worst-case error (for instance `spray1`) was within 3 cm, demonstrating the robustness of the system. Since S-TIQF was a category-level pose estimation method, these objects were novel and unseen. Hence, the ADI error was also proportional to the reconstruction accuracy seen from Fig. 5.12. For instance, the `wineglass2` had higher reconstruction error (CD $\approx$ 3 cm) resulting in lower pose estimation accuracy ($\texttt{err}_{adi} < 3$ cm). Furthermore, from Fig. 5.17 it can be seen that the proposed method compared favourably with state-of-the-art methods for category-level pose estimation using NOCS based reconstruction by improving the median ADI error by 35% and 20% for sparse tactile and dense visual point clouds respectively ($p < 0.001$). This shows that the proposed method was not tuned to a particular shape-reconstruction technique but was also adaptable to other category-level reconstruction techniques.

5.4.2 Overall System Evaluation

To evaluate the performance of the entire pipeline as shown in Fig. 5.1, the criterion was if the pose estimation error $\texttt{err}_{adi} < 3$ cm for the target object, then the experimental

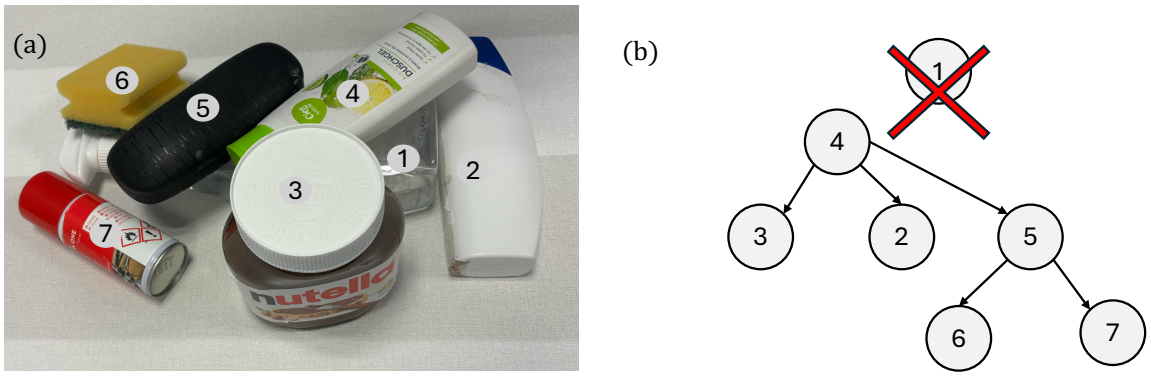


Figure 5.22: Failure case in generating the declutter scene graph as the target object (object 1) was not identified due to extremely dense clutter.

trial was deemed successful. This criterion was more competitive than other state-of-the-art approaches that allowed 5 cm error for category-level pose estimation even without overlapping clutter (Wang et al., 2019). The overall success rate was 88%.

Although the reconstruction and category-level pose estimation parts of the pipeline produced $\texttt{err}_{adi} < 3$ cm for all objects, failure cases were due to the interactive decluttering part (Fig. 5.2a). The reasons for failure cases include: (i) selection and execution of incorrect grasp/push poses and (ii) extremely dense clutter where less than 10% of the target object area was visible to the camera (as seen in Fig. 5.22). If the decluttering actions failed, then the human user intervenes and resets the scene. Complex objects were used to clutter the target object that included deformable (sponge) and transparent objects (triangle box) (see Fig. 5.10b). Furthermore, as an off-the-shelf semantic segmentation network (Chen et al., 2017) was used for extracting the scene graph and for grasp/push pose prediction, it could sometimes produce erroneous segmentation outputs for extreme dense clutter scenarios. As this was not the main contribution of this thesis, it may be substituted with state-of-the-art segmentation models such as Segment Anything Models (Kirillov et al., 2023) to improve the performance and can be considered as part of future work.

In summary, this chapter presented a novel approach for shared visuo-tactile-based category-level reconstruction and pose estimation of unknown target objects in dense clutter. Two robots equipped with vision and tactile sensors coordinated to declutter the workspace using the declutter scene graph approach. Visual and tactile sensing data were efficiently shared to explore the unknown target object using the joint information gain criteria. This ensured non-redundant actions performed in a greedy-information gain manner. Tactile perception was prioritised for transparent objects which were challenging for visual perception. The extracted point cloud data was used for inferring the reconstructed model of the object using the proposed reconstruction network. Finally, the novel S-TIQF

method was performed for robust pose estimation that was accurate for both sparse and dense point clouds. It provided globally optimal pose that was robust against local minima. The proposed method has been extensively validated using benchmark datasets and with real-robot experiments and it outperformed state-of-the-art techniques. Furthermore, the S-TIQF method can also be used for hand-eye calibration using any arbitrary objects through visual-tactile data which was critical for shared multi-modal perception.

Considering real-world applications, the proposed S-TIQF method can be applied in bin-picking applications in manufacturing industries for the retrieval of objects in unstructured clutter. Moreover, these objects may also have shiny and reflective surfaces or regions of transparency where tactile sensing can be useful. The safe manipulation of fragile objects requires an accurate pose estimate as a prerequisite and monitoring of contact forces during manipulation actions. The interactive perception methods in which robots can dynamically choose grasp or push actions to rearrange objects can be used for autonomous robotic assembly of complex objects. The S-TIQF sparse-to-dense point cloud registration approach can also be applied in industrial quality control analyses where sensor-acquired sparse point clouds of manufactured objects need to be aligned with dense volumetric point clouds derived from CAD models.

Chapter 6

Visuo-Tactile Goal-Driven Manipulation and Tracking of Novel Articulated Objects

Parts of this chapter are under preparation for submission as:

- "*Visuo-Tactile Goal-Driven Manipulation and Tracking of Articulated Objects*," **P. K. Murali**, B.Porr, and M. Kaboli.

The video of the experiments from this chapter is available here: `https://drive.google.com/file/d/1Ae80ZCKvD8yjvdHgcJ5SjFzY3nvUb4Km/view?usp=sharing`

6.1 Introduction

For robots to work in dynamic environments both perceiving and manipulating complex objects are often necessary. Articulated objects are ubiquitous in our surroundings exemplified by items such as cabinet drawers, eyeglass frames, scissors, laptops, and so on. Real-time pose tracking of articulated objects is a challenging task attributable to the high dimensionality of the state space. This stems from the multiple degrees of freedom and the inherent nonlinearity in dynamics, wherein the motion imparted to one component nonlinearly influences the movement of interconnected parts. Furthermore, interactive perception, wherein purposeful manipulation actions are performed to improve perception, is often necessary to detect possible articulation joints and correctly manipulate objects without damaging them (Bohg et al., 2017). Hence, there is a clear requirement for augmenting

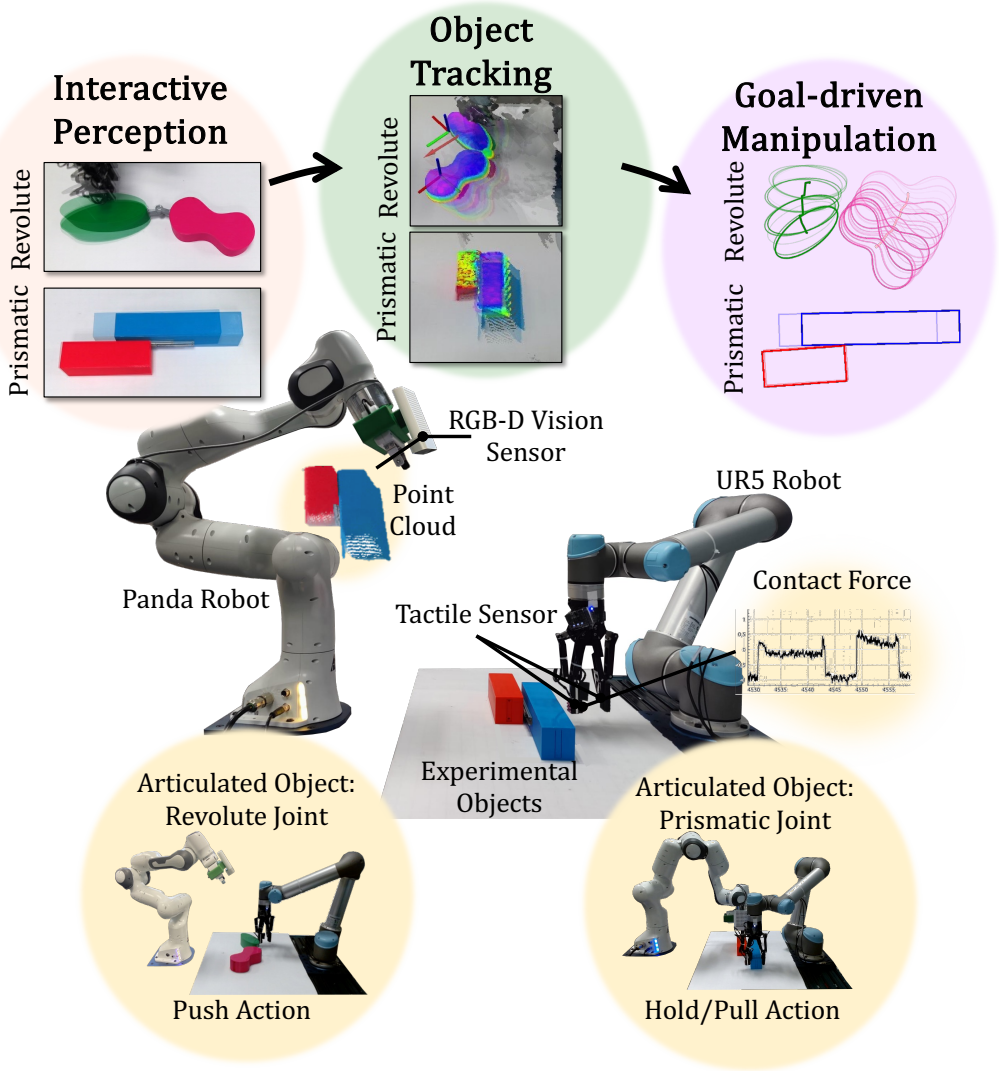


Figure 6.1: Experimental setup: A Franka Emika Panda robot with a Azure Kinect DK RGB-D vision sensor and a Universal Robots UR5 sensorised with tactile sensor arrays on the Robotiq Gripper, with unknown articulated objects in the workspace. The robots perform interactive perception to detect possible articulation structure in the objects. The objects are tracked using the proposed ArtReg algorithm and the 6 degree-of-freedom (DoF) tracking information is used for goal-driven manipulation.

visual perception with tactile sensing capable of discerning contact force and location. Pose tracking algorithms that can leverage multi-modal sensing data and operate without prior knowledge of object models or kinematic properties can enhance the versatility and adaptability of robotic systems.

Several prior works on the tracking of articulated objects have fundamentally relied upon geometric feature tracking or the utilisation of marker-based methodologies applied to sequential image data (Martín-Martín and Brock, 2022; Sturm et al., 2011). Other works

have also assumed prior knowledge of the object models or the kinematic structure of the articulated objects for tracking and manipulation (Nickels and Hutchinson, 2001; Paolillo et al., 2018). However, the predominant focus has been on visual inputs, overlooking the potential benefits of incorporating tactile information. Tactile information can provide complementary information regarding the object properties and is invariant to occlusions, ambient lighting, and object transparency (Li et al., 2020a). Furthermore, contact force information is useful during interactive perception to detect possible articulation and joint constraint limits without damaging the objects. Moreover, real-time tracking of unseen objects without prior knowledge regarding articulation constraints is an open research problem. Pose tracking enables further downstream tasks such as goal-driven manipulation. Most prior works performing closed-loop control for goal-driven manipulation used only rigid objects or disregarded proprioceptive or tactile information for the task (Lloyd and Lepora, 2021; Paolillo et al., 2018; Katz and Brock, 2008; Xu et al., 2022). Adapting such techniques directly to articulated objects presents a non-trivial undertaking, necessitating novel approaches tailored to the complexities inherent in articulated structures.

Tackling these current challenges in the state-of-the-art, a novel framework for visuo-tactile-based interactive perception for detecting, tracking and manipulating unknown novel objects (single, multiple, and articulated with revolute or prismatic joints) without assuming any prior knowledge regarding object shape or dynamics is presented.

The contributions of this chapter are as follows:

I ArtReg (Articulated Registration), a novel method for tracking unknown novel objects (single, multiple, or articulated) by integrating visual and tactile sensing in a Manifold Unscented Kalman Filter formulation on the SE(3) Lie Group.

II A novel method for the detection of kinematic chains in objects using a combination of push or hold-pull manipulation actions facilitated by autonomous interactive visuo-tactile perception.

III Leveraging the proposed ArtReg pose tracker, a visuo-tactile based closed-loop control algorithm has been designed intended for precise manipulation of objects to a goal configuration. The full-fledged framework operates effectively under a variety of conditions, including low illumination, visually complex backgrounds, and variations in the centre of mass of the objects.

Extensive experiments have been performed with two robot setup shown in Fig. 6.1 where the Panda robot was equipped with a RGB-D visual sensor and UR5 robot was sensorised with tactile sensor arrays on the gripper.

6.2 Methodology

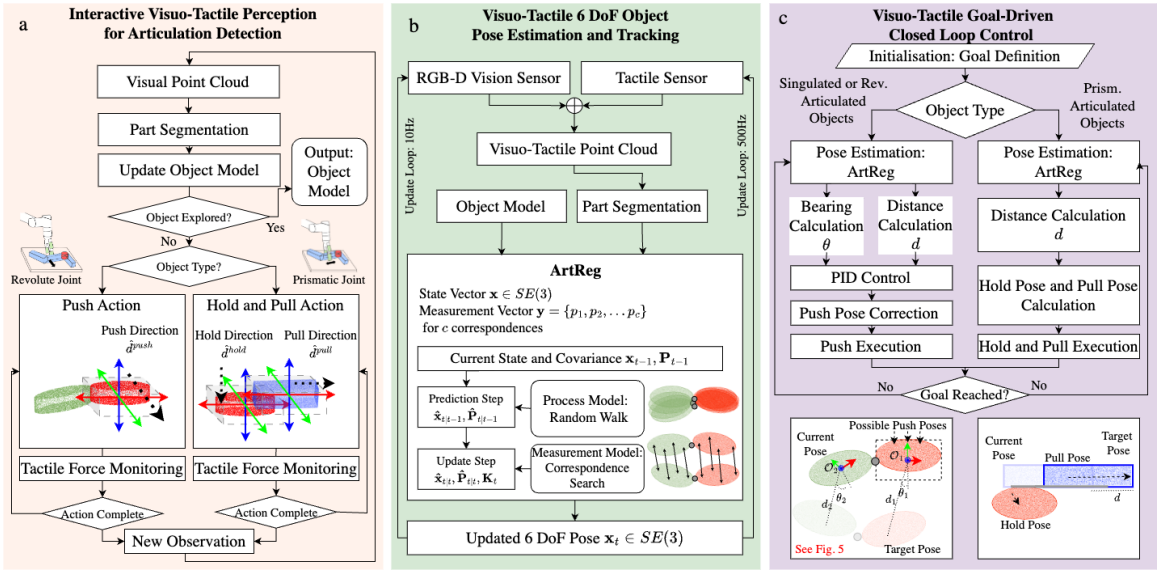


Figure 6.2: Outline of the proposed framework: (a) Interactive visuo-tactile perception for articulation detection, (b) Visuo-tactile-based pose tracking method termed ArtReg, and (c) Visuo-tactile-based closed-loop control for goal-driven manipulation.

6.2.1 Problem Formulation and Proposed Framework

A novel framework for detecting, tracking and goal-driven manipulation of novel unknown objects (single, multiple or articulated) without assuming any prior knowledge regarding object shape or dynamics is presented in this chapter as shown in Fig. 6.2. The two robot team shown in Fig. 6.1 autonomously performed interactive perception to detect possible articulation and infer its underlying kinematics if multiple objects were present in the workspace using pushing or hold-pull action manoeuvres as described in Fig. 6.2a and Sec. 6.2.2. A novel method termed ArtReg is presented for the accurate pose tracking of objects using manifold unscented Kalman filter shown in Fig. 6.2b and Sec. 6.2.3. The final part of the framework (Fig. 6.2c) presents a closed-loop goal-driven manipulation approach for manipulating articulated objects (with revolute or prismatic joints) or single objects to a desired goal-pose by relying upon visual and tactile sensing described in Sec. 6.2.4.

6.2.2 Interactive Perception for Articulation Detection

Interactive perception techniques and the proposed pose tracker, ArtReg have been used in conjunction for distinguishing articulated objects from rigid objects as well as identifying the type of articulation joints present. The degree of articulation was limited to 1 DoF

revolute and prismatic joints in this study. However, the proposed method could be trivially extended for n-DoF systems. The visual point clouds of the objects were segmented using the proposed part segmentation approach to obtain the estimated number of parts present and possible articulation joint (object kinematic model). An iterative interactive perception approach was designed which updates the belief regarding the object kinematic model with each manipulation action as shown in Fig. 6.2a. These actions which were *push* or *hold-pull* were chosen and performed autonomously by the robot. The actions were chosen in greedy manner, which allowed extracting the correct object kinematic structure with minimal number of actions. The proposed part segmentation approach, action selection, and execution have been detailed in this section.

Part Segmentation

Objects on a planar surface in the kinematic limits of the robot were to be segmented from the background and separated into different parts depending on the object configuration. Consider the Fig. 6.2a wherein from the RGB point cloud of the scene extracted from the visual camera, firstly the background plane was segmented using the RANSAC-based plane fitting method. All points within a user-defined distance threshold of a best-fitting plane were filtered out from the visual point cloud. Secondly, during interactive perception the robot was also in contact with the object of interest and possibly occluded the object. Hence, it was necessary to also segment the robot links from the point cloud. Given the current joint pose information of the robot body which was obtained from the robot kinematic model, the robot mesh was projected onto the point cloud, and the 3D points which correspond to the robot body are filtered out from the point cloud. Thirdly, the remaining point cloud which consisted of the objects and possibly noisy outlier points were filtered out using statistical outlier removal methods that removed points further away from neighbourhood points compared to the average for the point cloud. Finally, there may be multiple objects present in the scene with optional kinematic links between them. Hence, to perform part segmentation, the region-growing segmentation algorithm (Rusu et al., 2008) was performed. Provided with an initial seed point, each neighbourhood point was checked if it belonged to the same region as the seed point or if it should be considered as a future seed point in a future iteration. The criteria for inclusion in the same region depend on a smoothness constraint (based on surface normals) and colour information. After an initial segmentation process, regions having similar colour information were merged together. An implementation of the region growing algorithm available in the Point Cloud Library (PCL) (Rusu and Cousins, 2011) was used. As described in the framework in Fig. 6.2a, after each action was performed, the part segmentation provided the belief of the number

of objects in the scene which was updated upon subsequent actions. The object was considered to be explored completely if the belief on the object kinematic model does not change with 3 consecutive iterations.

Interactive Perception Action Selection

Referring to Fig. 6.2a, once the part segmentation was performed, the object type (prismatic, revolute articulated or rigid object) was estimated by the robots autonomously interacting with the objects. Two types of actions were defined: *push* and *hold-pull* actions. The push action was used to distinguish if an object is rigid or articulated with a revolute joint. The hold-pull action was only used to distinguish whether an object has a prismatic joint. Consider the Fig. 6.3 which describes the action execution procedure for the object type detection. The object detection strategy is as follows: if the current belief from the part segmentation had multiple objects, then the robot always began with pushing action. The object(s) were tracked using ArtReg, and the change in relative pose after performing the action was used to infer the joint type. The heuristic for joint-type inference is as follows: if the object pushed one part and there was a change in pose registered for both parts, then its was likely to have a rigid joint (Fig. 6.3a rigid joint). In that case, the robot proceeded to push the other part and if a similar behaviour was observed, then the belief was confirmed to be either a rigid object or an object with prismatic joint. In contrast, if only the pushed part was moved while the remaining parts were static, it is likely that the parts were connected to a revolute joint. The robot pushed the other part consequently to confirm the object kinematic structure. This process can be seen in Fig. 6.3a. If pushing action was not possible to distinguish the object kinematics conclusively, the system reverts to performing hold-pull action sequence (Fig. 6.3b). One robot holds one part by applying an orthogonal force to the surface of the object and another robot tries to grasp and pull the other part. If a change in pose was detected after performing the action, it could be inferred that the parts were linked through a prismatic joint. This process can be seen in Fig. 6.3b.

Action Execution

For both types of predefined actions i.e., push and hold-pull, the action parameter selection was based on geometric information extracted from the point cloud. The oriented bounding box (OBB) of the point cloud representing the object was extracted. Consider the Fig. 6.3a wherein the eigenvectors $(\mathbb{V}_x, \mathbb{V}_y, \mathbb{V}_z)$ were extracted from the covariance matrix of the point cloud representing the object. The major eigenvector $\mathbb{V}_x$ represents the axis along the length of the object, the middle eigenvector $\mathbb{V}_y$ is perpendicular to the object and the minor

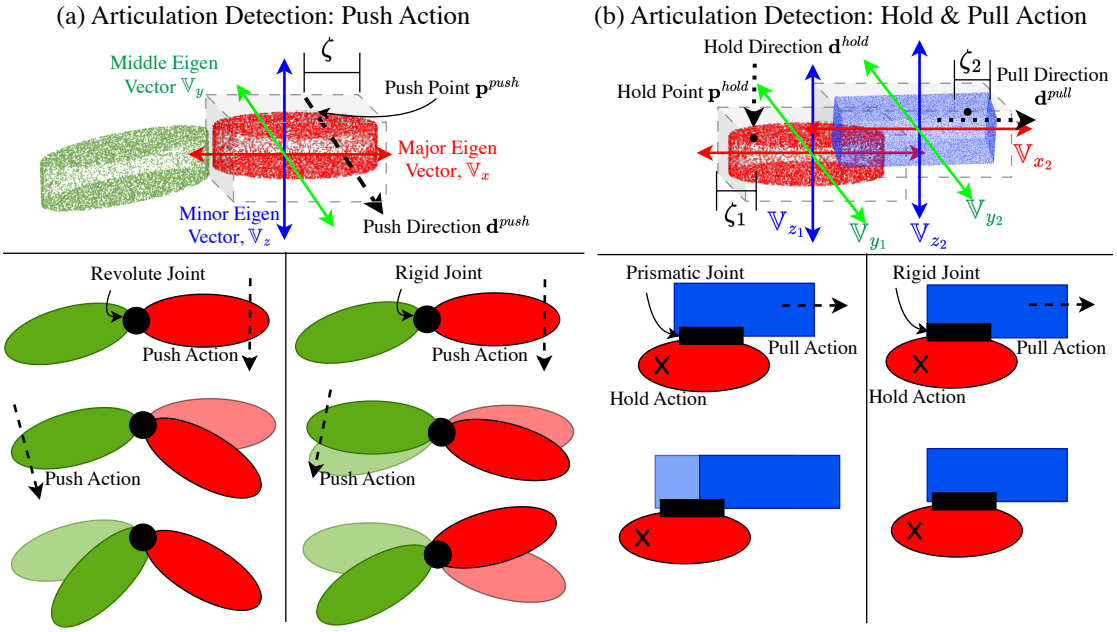


Figure 6.3: Interactive perception for articulation detection: (a) push action for revolute joints, (b) grasp and pull action for prismatic joints.

eigenvector $\mathbb{V}_z$ is normal to the surface of the object. Push action is parameterized by a push position and direction, $a^{push} = \{\mathbf{p}^{push}, \mathbf{d}^{push}\}$. For detecting articulated objects, the push direction was parallel to $\mathbb{V}_y$ and the push position was along the face of the oriented bounding box that was parallel to $\mathbb{V}_y$ with a tolerance distance ζ such that the push direction was intersecting along the edge of the object. The pushing distance was predefined for 10 cm.

In contrast, for the hold-pull action, both robots were used similarly to how humans perform bimanual manipulation actions to pull apart two parts linked by a prismatic joint. Referring to Fig. 6.1, the Panda robot which was equipped with the RGB-D vision camera but without tactile sensing was used for performing the hold action. The Panda robot also has force/torque sensors on all joints which were used for proprioceptive information. The UR5 robot which was sensorised with tactile sensing was used to perform the grasp and pull action. In order to ensure that the robots do not collide with each other, the actions were performed close to the diametrically opposing edges of the object. The hold and pull actions were performed on the two different parts of the object. Referring to Fig. 6.3b, the hold action is parameterized by the hold position and direction as $a^{hold} = \{\mathbf{p}^{hold}, \mathbf{d}^{hold}\}$. Given two parts of the object denoted by 1 and 2, respectively, the hold direction $\mathbf{d}^{hold}$ is parallel to the minor eigenvector for the one of the object parts as $\mathbb{V}_{z_1} \parallel \mathbf{d}^{hold}$. The hold position is obtained by moving the centroid of the part point cloud to the edge of

the bounding box with a certain tolerance distance ζ from the bounding box edge. The force-torque sensor on the joint 7 of the Panda robot was used to detect contact and apply a constant holding force on the object part. Similarly, the pull action is parameterized by position and direction $a^{pull} = \{\mathbf{p}^{pull}, \mathbf{d}^{pull}\}$. In order to grasp objects of different shapes, when provided with a position $\mathbf{p}^{pull}$, the robot was moved to a position vertically above the $\mathbf{p}^{pull}$ at a predefined height and the gripper opened to the maximum width. The robot moved in a straight-line downwards to the $\mathbf{p}^{pull}$ and closed the gripper until contact was detected by the tactile sensors. The gripper applied a constant grasping force on the object part. The position $\mathbf{p}^{pull}$ was obtained by shifting the centroid of the object part 2 along $\mathbb{V}_{x_2}$ to the edge of the bounding box with a certain tolerance distance ζ from the edge. The pull direction is parallel to the major eigenvector of the other object part such that $\mathbb{V}_{x_2} \mid\mid \mathbf{d}^{pull}$. Once the robot grasped the part of the object, it pulled for a predefined distance of 5 cm. The tactile sensors on the inner side of the gripper fingers were used to detect possible loss of contact and stop the robot.

6.2.3 Articulated Registration (ArtReg) for Visuo-Tactile Pose Tracking

The 6DoF pose was estimated through a Manifold Unscented Kalman Filter (UKF) on the SE(3) Lie Group. Note that the complete 6DoF pose of a dynamic object evolves on a SE(3) Lie Group. The derivation of a standard UKF on Euclidean space is detailed in Appendix A.3.2.

Manifold UKF on SE(3) Lie Group

Consider a system with known initial mean and initial covariance as $\bar{\mathbf{x}}_0 = E[\mathbf{x}_0]$ and $\mathbf{P}_0 = P[\mathbf{x}_0]$ respectively. The aim is to obtain the *a posteriori* state estimate $\hat{\mathbf{x}}_{k|k}$ and *a posteriori* covariance $\hat{\mathbf{P}}_{k|k}$ upto time k integrating k observations. The state $\hat{\mathbf{x}}_{k|k} \in SE(3)$ denotes the 6 DoF pose of the object. The state transitions model (also called process model) is denoted as $f(\cdot)$ and the measurement model as $h(\cdot)$. The *a posteriori* state can be obtained as follows:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}(\mathbf{z}_{k|k} - \hat{\mathbf{z}}_{k|k-1}) \tag{6.1}$$

wherein, $\hat{\mathbf{x}}_{k|k}$ is the *a posteriori* state variable, $\mathbf{K}$ is the Kalman gain, $\hat{\mathbf{x}}_{k|k-1}$ refers to predicted *a priori* state variable, $\mathbf{z}_{k|k}$ is the measurement observation at time step k and $\hat{\mathbf{z}}_{k|k-1}$ is the predicted measurement observation from the a priori state information.

In the case of the manifold UKF, the state $\mathbf{x}$ which describes the pose in SE(3) evolves in the manifold $\mathcal{M}$ as described in Fig. 6.4. The tangent space at $\mathbf{x}$ is defined as $T_{\mathbf{x}}\mathcal{M}$. There

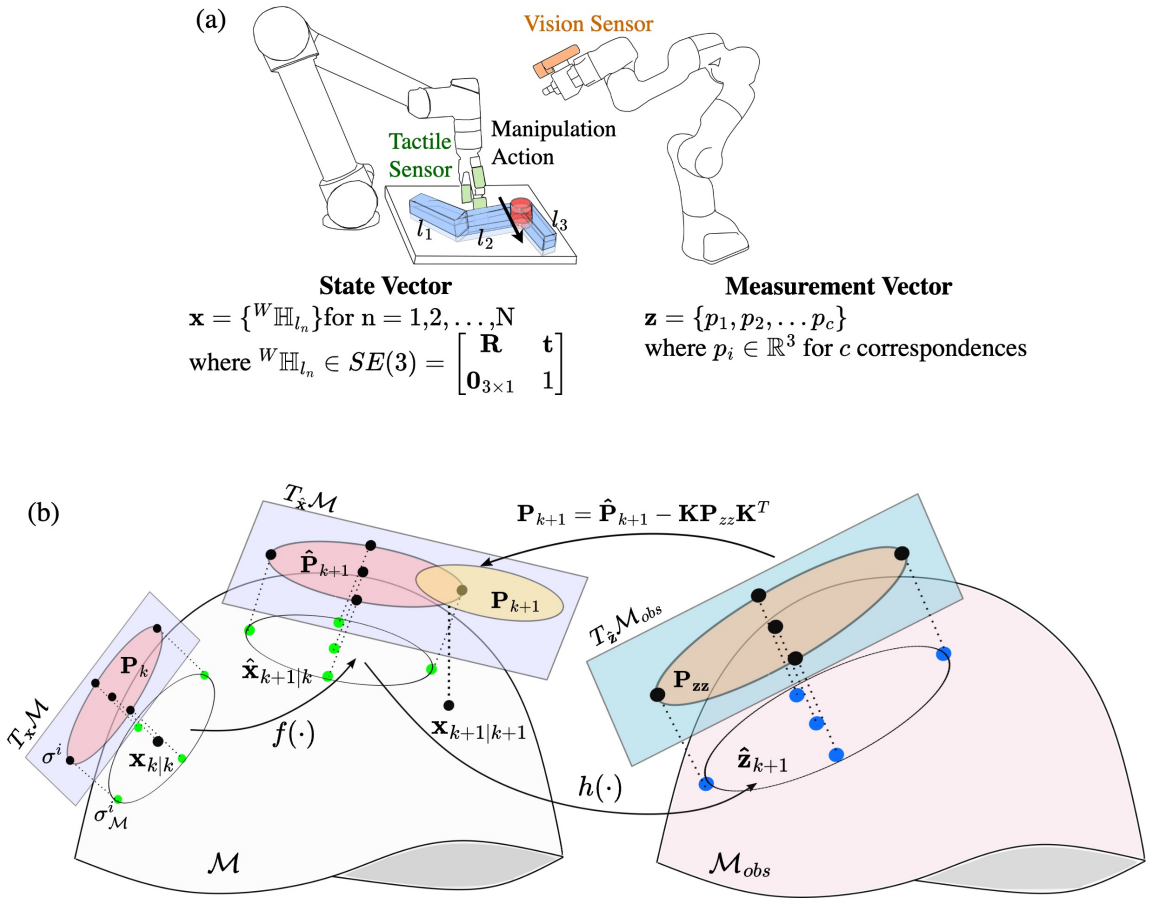


Figure 6.4: (a) The experimental setup shown along with the description of the state and measurement vector. (b) The proposed ArtReg filter which is a manifold unscented Kalman filter visualised with operations on the state manifold $\mathcal{M}$ and measurement manifold $\mathcal{M}_{obs}$.

are two sets of sigma points describing the covariance $\mathbf{P}$ in the manifold and the tangent space i.e., $\sigma^i \in T_{\mathbf{x}}\mathcal{M}$ and $\sigma^i_{\mathcal{M}} \in \mathcal{M}$ wherein $i = 0, 1, \dots, 2M$. The sigma points on the tangent space are the same as described in the Euclidean space (see Appendix A.3.2, Eq. (A.22)). The mapping $\varphi(\cdot)$ also called as the *retraction* transforms a point in the tangent space into the manifold space and $\varphi^{-1}(\cdot)$ performs the inverse operation. In the SE(3) Lie Group, $\varphi(\cdot)$ is also called as the exponential map (exp) that maps the elements of the Lie algebra to the Lie Group and $\varphi^{-1}(\cdot)$ is the logarithmic map (Log). A rigorous introduction to the concept of Lie theory is beyond the scope of this work and the readers are advised to refer to (Sola et al., 2018). The prediction and update steps of the manifold UKF are described below and the schematic illustration is provided in Fig. 6.4b.

Prediction Step: The mean is propagated using the non-noisy process model $f(\cdot)$:

$\mathcal{M} \rightarrow \mathcal{M}$ and input vector u_k as:

$$\hat{\mathbf{x}}_{k|k-1} = f(\hat{\mathbf{x}}_{k-1|k-1}, u_k, \mathbf{0}) \tag{6.2}$$

Contrary to the classical Euclidean UKF, wherein the weighted mean of the sigma points are propagated through the process model, in the manifold UKF, propagating the state directly through the process model avoids the computational complexity of finding the weighted averaging on manifold $\mathcal{M}$ (Brossard et al., 2020). To compute the covariance $\mathbf{P}_{k|k-1}$, the sigma points are propagated through the process model as:

$$\mathbf{P}^{s}_{k|k-1} = \sum_{i=1}^{2M} W_i \varphi^{-1}_{\hat{\mathbf{x}}_{k|k-1}}(f(\sigma^{i}_{\mathcal{M}}))(\varphi^{-1}_{\hat{\mathbf{x}}_{k|k-1}}(f(\sigma^{i}_{\mathcal{M}})))^T \tag{6.3}$$

Similarly, the noise sigma points are also propagated through the process model and added to the covariance as:

$$\mathbf{P}^{n}_{k|k-1} = \sum_{i=1}^{2M} W_i \varphi^{-1}_{\hat{\mathbf{x}}_{k|k-1}}(f(\sigma^{'i}_{\mathcal{M}}))(\varphi^{-1}_{\hat{\mathbf{x}}_{k|k-1}}(f(\sigma^{'i}_{\mathcal{M}})))^T \tag{6.4}$$

wherein $\sigma^{'i} = \pm(\sqrt{(M+\lambda)\mathbf{Q}})_i$. The covariance is calculated as:

$$\mathbf{P}_{k|k-1} = \mathbf{P}^{s}_{k|k-1} + \mathbf{P}^{n}_{k|k-1} \tag{6.5}$$

Update Step: The measurement model $h(\cdot)$ generates an predicted observation for each sigma point. The predicted observation $\hat{\mathbf{z}}_{k|k-1}$ belongs to the observation manifold $\mathcal{M}_{obs}$ such that $h(\cdot) : \mathcal{M} \rightarrow \mathcal{M}_{obs}$. The predicted measurements are calculated using sigma points on the manifold:

$$\hat{\mathbf{z}}_{k|k-1} = \sum_{i=0}^{2M} W_i^{(m)} h(\sigma^{i}_{\mathcal{M}}) \tag{6.6}$$

The covariance and the cross-covariance are calculated similarly using the sigma points on the manifold as:

$$\begin{aligned} \mathbf{P}_{zz} &= \sum_{i=0}^{2M} W_i^{(c)} (h(\sigma^{i}_{\mathcal{M}}) - \hat{\mathbf{z}}_{k|k-1})(h(\sigma^{i}_{\mathcal{M}}) - \hat{\mathbf{z}}_{k|k-1})^T + \mathbf{R}_k \\ \mathbf{P}_{xz} &= \sum_{i=1}^{2M} W_i^{(c)} \sigma^{i}_{\mathcal{M}} (h(\sigma)^{i}_{\mathcal{M}} - \hat{\mathbf{z}}_{k|k-1})^T \end{aligned} \tag{6.7}$$

Once the predicted covariance and cross-covariance is calculated, the Kalman gain is computed as:

$$\mathbf{K}_k = \mathbf{P}_{xz}\mathbf{P}_{zz}^{-1} \tag{6.8}$$

Note that the covariance and cross-covariance lies on the tangent spaces $T_{\hat{\mathbf{x}}}\mathcal{M}$ and $T_{\hat{\mathbf{z}}}\mathcal{M}_{obs}$ respectively. The updated state is calculated using the retraction back onto the manifold as:

$$\hat{\mathbf{x}}_{k|k} = \varphi(\hat{\mathbf{x}}_{k|k-1}, \mathbf{K}(\mathbf{z}_{k|k} - \hat{\mathbf{z}}_{k|k-1})) \tag{6.9}$$

The *a posteriori* covariance is calculated from the predicted covariance and cross-covariance as follows

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k\mathbf{P}_{zz}\mathbf{K}_k^T \tag{6.10}$$

The process is shown graphically in Fig. 6.4b.

In general, for any arbitrary N objects that need to be tracked, state variables $\mathbf{x}^i$ for $i = 1, 2, \ldots, N$ can be defined which are tracked independently using N manifold UKFs. The formulation is general and can be deployed for single, multiple or articulated object tracking in an identical manner. The framework for ArtReg is described in Fig. 6.2b. For a given point cloud of objects captured by the camera, the number of objects to be tracked was provided by the segmentation method described in Sec. 6.2.2. If tactile contact points are available during robot interaction, these are added to the observed point cloud. For each object $O^{i=1:N}$, the segmented point cloud at the initial state at time $t = 0$ corresponds to the object point cloud $\mathcal{O}^{i=1:N}$. At each time instant k, the objective is to track the pose $\mathbf{x}_{k|k}^i \in SE(3)$ of O_i given the previous state information $\mathbf{x}_{k|k-1}^i$ and measurements $\mathbf{z}_k^i$. For the sake of simplicity of notations, the pose tracking process is described here for a single object O. Extending the procedure for N objects is trivial as the states $\mathbf{x}^i$ are independent of each other for $i \in [1, N]$.

The process model $f(\cdot)$ is defined as a random walk model as $\hat{\mathbf{x}}_{k|k-1} = f(\hat{\mathbf{x}}_{k-1|k-1}, u_k, w_k)$ wherein the input vector $u_k = \mathbf{0}$ and noise model $w_k \sim \mathcal{N}(0, \mathbf{Q})$ represents a zero-mean Gaussian noise. The state is constant and the random walk is induced through the state noise according to Eq. 6.2-6.5.

The measurement model $h(\cdot)$ defines the expected measurement $\hat{\mathbf{z}}_{k|k-1}$ when the object O is at the predicted state $\hat{\mathbf{x}}_{k|k-1}$. Let the point cloud extract at time step k be denoted as $\mathcal{S}_k^{full}$. Part segmentation is performed to extract the various objects present in the scene $\mathcal{S}_k^j$ for $j = 1 : L$ where in ideal cases $L = N$ and in noisy conditions or close overlap between objects $L \neq N$. The measurement vector consists of 3D points (x, y, z) belonging to object $\mathcal{S}^j$ i.e., $\mathbf{y}_{k|k} = \{p_1, p_2 \ldots p_Q\} \forall p \in \mathcal{S}_k^j$. The correspondence search finds the closest

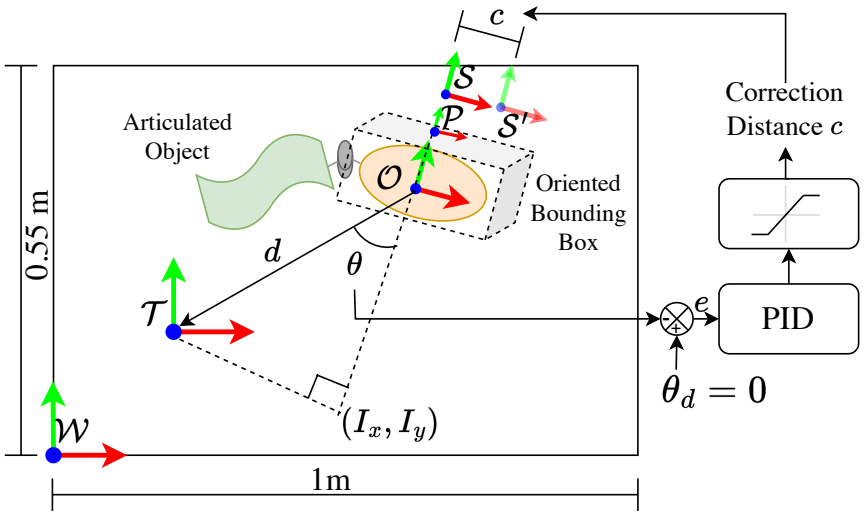


Figure 6.5: Goal-driven closed loop control system

corresponding points in $\mathcal{O}$ and $\mathcal{S}^j$ i.e.,

$$\xi(\sigma_{\mathcal{M}}) = argmin_{p \in \mathcal{O}}(||\mathcal{S}^j - \sigma_{\mathcal{M}} p||) \tag{6.11}$$

The 3D points belong to the point cloud of the object $\mathcal{O}$ are transformed to the predicted pose for each sigma point $\sigma_{\mathcal{M}} \in SE(3)$. Since there could be multiple objects in the scene such that $\mathcal{S}^{j=1:L}$, the correspondence search ensures to find the object with the closest Euclidean distance compared with the predicted pose of the object. The measurement model provides the correspondence points in the object point cloud with respect to the measured scene point cloud such that $h(\sigma_{\mathcal{M}}^{(m=0:2M)}) = \xi(\sigma_{\mathcal{M}}^{(m=0:2M)})$. The *a posteriori* state is obtained by minimising the distance between the correspondence points between the object point cloud and the scene point cloud $(\mathbf{z}_{k|k} - \hat{\mathbf{z}}_{k|k-1})$ using Eq. 6.9. The state and measurement noise ($\mathbf{Q}$ and $\mathbf{R}$ respectively) are initialised as diagonal matrices with random values. The ratio between the state and measurement noise determines if the filter tracks the predicted states (if state noise is lower) or if it tracks the measurements (if the measurement noise is lower) (Welch and Bishop (1995): Pg 32-35). Since the process models resembles a random walk, the measurement noise is initialised with a value lower than the state noise in order to track the dynamically moving object(s). The overall algorithm is shown in Algorithm 4.

6.2.4 Visuo-Tactile Goal-driven Closed-Loop Control

The objective of the closed-loop control is to push the object (single or articulated) to a pre-defined goal pose. While the closed-loop controller is identical for both single

Algorithm 4: ArtReg: Object tracking with Manifold Unscented Kalman Filter

Input: $\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}, \mathbf{R}_{k-1}, \mathbf{Q}_{k-1}, u_k$
Output: $\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k}, \mathbf{K}_k$
Prediction:
//Propagate the mean state
$\hat{\mathbf{x}}_{k|k-1} = f(\hat{\mathbf{x}}_{k-1|k-1}, u_k, \mathbf{0})$
//Compute the sigma points on tangent space and manifold
$\sigma^i, \sigma^i_{\mathcal{M}}$ for $i = 0, \ldots, 2M$ where $M = 6$. //State Covariance and Noise Propagation
$\mathbf{P}^s_{k|k-1} = \sum_{i=1}^{2M} W_i \varphi^{-1}_{\hat{\mathbf{x}}_{k|k-1}}(f(\sigma^i_{\mathcal{M}}))(\varphi^{-1}_{\hat{\mathbf{x}}_{k|k-1}}(f(\sigma^i_{\mathcal{M}})))^T$
$\mathbf{P}^n_{k|k-1} = \sum_{i=1}^{2M} W_i \varphi^{-1}_{\hat{\mathbf{x}}_{k|k-1}}(f(\sigma'^i_{\mathcal{M}}))(\varphi^{-1}_{\hat{\mathbf{x}}_{k|k-1}}(f(\sigma'^i_{\mathcal{M}})))^T$
$\mathbf{P}_{k|k-1} = \mathbf{P}^s_{k|k-1} + \mathbf{P}^n_{k|k-1}$
Update:
$\mathbf{z}_{k|k}$; ▷ Get actual measurement from robot
//Compute measurement sigma points
$\hat{\mathbf{z}}_{k|k-1} = \sum_{i=0}^{2M} W_i^{(m)} h(\sigma^i_{\mathcal{M}})$
//Compute covariance and cross-covariance matrix
$\mathbf{P}_{zz} = \sum_{i=0}^{2M} W_i^{(c)} (h(\sigma^i_{\mathcal{M}}) - \hat{\mathbf{z}}_{k|k-1})(h(\sigma^i_{\mathcal{M}}) - \hat{\mathbf{z}}_{k|k-1})^T + \mathbf{R}_k$
$\mathbf{P}_{xz} = \sum_{i=1}^{2M} W_i^{(c)} \sigma^i_{\mathcal{M}} (h(\sigma)^i_{\mathcal{M}} - \hat{\mathbf{z}}_{k|k-1})^T$
//Compute Kalman Gain
$\mathbf{K}_k = \mathbf{P}_{xz}\mathbf{P}^{-1}_{zz}$
//Compute Mean and Covariance update
$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{P}_{zz} \mathbf{K}_k^T$
$\hat{\mathbf{x}}_{k|k} = \varphi(\hat{\mathbf{x}}_{k|k-1}, \mathbf{K}(\mathbf{z}_{k|k} - \hat{\mathbf{z}}_{k|k-1}))$

or articulated revolute objects, in the case of articulated objects a task planner decides which part of the articulated object to push based on the distance to goal. Without loss of generality, the push controller is detailed for a single object as shown in Fig. 6.5. Definition of the coordinate frames used to devise the controller: the geometric centre of the oriented bounding box (OBB) is defined as $\mathcal{O}$; the target frame $\mathcal{T}$; pushing frame as $\mathcal{P}$; tactile sensor frame as $\mathcal{S}$ and world coordinate frame as $\mathcal{W}$. The objective is to align $\mathcal{O}$ with $\mathcal{T}$ within a defined tolerance error λ such that $d \to 0, \theta \to 0$. Given the current pose of the object as ${}^{\mathbf{W}}H_{\mathcal{O}}$, the push pose ${}^{\mathbf{W}}H_{\mathcal{P}}$ is computed along the middle eigenvector $\mathtt{V}_y$ such that it passes through the Geometric center $\mathcal{O}$ as detailed in Sec. 6.2.2. However, when pushing to a predefined goal, the push position $\mathbf{p}^{push}$ is shifted along the perimeter of the object such that the bearing (angular offset from the goal) is minimised. The push direction vector $\mathbf{d}^{push}$ is kept the same in order to ensure the maximum area of the tactile sensor array is in contact with the object. To minimise θ, the sensor frame is shifted along the object perimeter to $\mathbf{S}'$ based on the output from a proportional-integral-derivative (PID) controller. The angular

offset θ is measured as follows: The slope of the line $l_{\mathcal{PO}}$ passing through $(\mathcal{O}_x, \mathcal{O}_y)$ and $(\mathcal{P}_x, \mathcal{P}_y)$ is

$$m = \frac{\mathcal{P}_y - \mathcal{O}_y}{\mathcal{P}_x - \mathcal{O}_x} \tag{6.12}$$

The perpendicular projection of the point $(\mathcal{T}_x, \mathcal{T}_y)$ on the line $l_{\mathcal{PO}}$ has a negative reciprocal slope $-1/m$. The equation of perpendicular projection from $\mathcal{T}$ is $(y - \mathcal{T}_y) = \frac{-1}{m}(x - \mathcal{T}_x)$. Solving for the point of intersection (I_x, I_y):

$$\begin{aligned} I_x &= \frac{\mathcal{T}_x - m^2\mathcal{O}_x - m(\mathcal{O}_y - \mathcal{T}_y)}{1 + m^2} \\ I_y &= (-1/m)I_x + \mathcal{T}_y \end{aligned} \tag{6.13}$$

The angle θ is computed as:

$$\theta = \tan^{-1}\left(\frac{\sqrt{(\mathcal{T}_x - I_x)^2 + (\mathcal{T}_y - I_y)^2}}{\sqrt{(\mathcal{O}_x - I_x)^2 + (\mathcal{O}_y - I_y)^2}}\right) \tag{6.14}$$

The input to the PID control is $e = \theta - \theta_d$ where $\theta_d = 0$. The PID control output at time t is provided as:

$$c(t) = k_p e(t) + k_i \int_0^t e(\tau) d\tau + k_d \frac{de}{dt} \tag{6.15}$$

The PID gain coefficients k_p, k_i, k_d were computed empirically through trail-and-error as 0.05, 0.0, 0.03 respectively. Ad-hoc tuning of the controller was performed due to the relatively simple control problem. The controller output is passed through a saturation function wherein the output value c is clipped if it is beyond ± 3 cm. The correction distance c is incorporated into the sensor pose as:

$${}^{\mathcal{W}}H_{\mathcal{S}'} = {}^{\mathcal{W}}H_{\mathcal{S}}\,{}^{\mathcal{S}}H_{\mathcal{S}'} \tag{6.16}$$

wherein ${}^{\mathcal{S}}H_{\mathcal{S}'} = \begin{bmatrix} & & c \\ & \mathbb{I}_3 & 0 \\ & & 0 \\ \mathbf{0}_{3x1} & & 1 \end{bmatrix}$. The distance to goal-pose is given as:

$$d = \sqrt{(\mathcal{T}_x - \mathcal{O}_x)^2 + (\mathcal{T}_y - \mathcal{O}_y)^2} \tag{6.17}$$

The pushing interaction was performed as a discrete iterative process wherein the robot performed the push action with 3 cm pushing distance. Tactile sensors were continuously monitored throughout the interaction to identify any instances of contact loss with the ob-

ject. As seen from Fig. 6.2c, the process of segmentation, pose estimation, point point and direction estimation and push execution was repeated until the object was aligned with the target pose with the user-specified tolerance bounds. The iterative pushing strategy allows the robot to transition between distinct segments of the articulated object such that the distance to goal for each segment was minimised uniformly.

6.3 Experimental Results

6.3.1 Experimental Setup and Outline

The experimental setup shown in Fig. 6.1 consists of a Universal Robots UR5 with the Robotiq gripper that is sensorised with tactile sensor arrays from Xela and Contactile sensors and a Franka Emika Panda robot equipped with a RGB-D Azure Kinect DK vision sensor attached with a custom flange to the end-effector as detailed in Chap. 3. Two different tactile sensors having different operational principles were chosen to demonstrate that the method was agnostic to any particular type of tactile sensor. All manipulation operations were performed on the robot workspace of size (1.0 m, 0.55 m). The workspace limits also roughly match the kinematic limits of the UR5 robot which has a maximum kinematic reach of 0.85 m. All experiments were performed on an Ubuntu 18.04 workstation with Intel Xeon Gold 5222 CPU. ROS Melodic was used as robotic middleware, Point cloud library (Rusu and Cousins, 2011) for operations involving point clouds, `kalmanif` libraries (Deray and Solà, 2020) for Kalman filter implementations.

Experimental Objects: The experimental objects are shown in Fig. 6.6. The naming convention used is the `colour-shape` of the object. The objects have been 3D printed with varying sizes and shapes. The following shapes were used: `cube`, `cuboid`, `oval`, `butter`, `sine`, and `triangle`. The reverse side of all objects have 3 hollowed out cylinders (as seen in Fig. 6.6a - `pink-butter` and `blue-sine/ gray-cuboid`) in which external weights of 500g can be placed to offset the centre-of-mass (CoM) of the objects. The external weight can be placed in any of the 3 possible holes unknown to the robotic system by the user and results in 3 different variations for CoM for each object. The camera always sees the smooth 'top' surface of the objects, hence the determination of offset in CoM can be performed using tactile sensing alone. Each of the objects can be attached to another with a revolute joint using a specialised 3D printed hinge (Fig. 6.6b). Furthermore, prismatic joints (typically used in drawers) attached to the objects axially to form an articulated prismatic object (Fig. 6.6c) were also used. As a special case of the revolute joint shown as `blue-sine/ green-butter`, the object had an overlapping revolute joint such that the bottom object (`green-butter`) has a cylindrical protrusion and the top object (`blue-sine`)

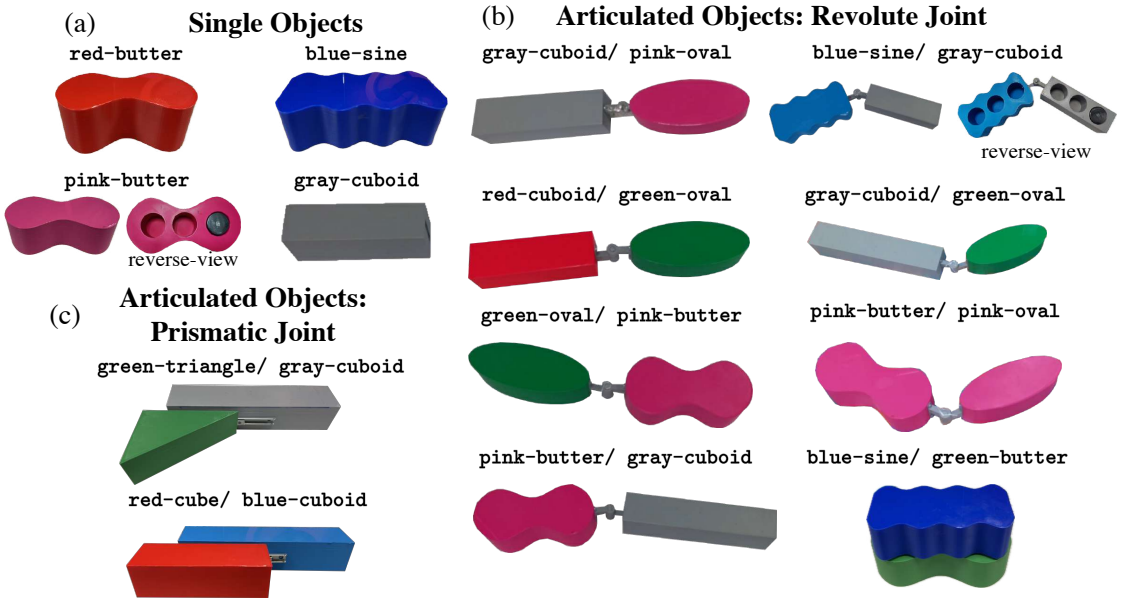


Figure 6.6: List of experimental objects used for tracking and closed-loop control: (a) single objects, (b) articulated objects with revolute joint and (c) articulated objects with prismatic joint. The reverse view of objects such as `blue-sine/ gray cuboid` and `pink-butter` is shown wherein the center of mass can be changed by placing an additional weight.

has the corresponding cylindrical hole allowing for a rotational joint of 1 DoF.

Outline of experiments: A comprehensive and exhaustive set of robotic experiments were performed to evaluate the entire pipeline. Detection of possible articulation as well as the type of articulated joints (revolute / prismatic) has been performed in Sec. 6.3.2. Closed loop control for goal-driven manipulation of singular and articulated objects were performed in Sec. 6.3.3. Furthermore, to evaluate for robustness, the goal-driven manipulation experiments were performed under various conditions such as (a) low lighting, (b) challenging backgrounds, and (c) varying CoM. In Sec. 6.3.4 only pose tracking was evaluated without any target-driven robot manipulation for single, multiple and articulated objects. The comparison with state-of-the-art approaches in demonstrated in Sec. 6.3.5 using a standard benchmark of synthetic articulated objects from the PartNet-Mobility dataset (Xiang et al., 2020).

6.3.2 Articulation Detection

For the detection of articulated objects from rigid objects, manipulation actions such as pushing and hold-pulling for interactive perception were used. In Fig. 6.7, two identical scenes were used wherein an `oval` object and `cuboid` object were connected with a revo-

lute joint (Fig. 6.7a-d) and a rigid joint (Fig. 6.7e-h) respectively. Since visual inspection alone cannot differentiate between the rigid object and the articulated object, interactive perception was employed. In Fig. 6.7a, the robot pushes the `cuboid` for a predefined distance of 10 cm and the ArtReg tracker showed that only the `cuboid` was displaced as evidenced in Fig. 6.7b. Subsequently, the robot pushed the `oval` object, resulting in its displacement while the connected `cuboid` remained stationary. This observation led to the conclusion that the object was articulated. On the contrary, in Fig 6.7e, the robot pushed the `oval` object for 10 cm and the `cuboid` object was also displaced. Similarly, the robot autonomously chose to push the `cuboid` object next which also moves the `oval` object proportionally. This showed that the objects were connected by a rigid joint and in fact was one rigid object. A similar experiment was performed with a configuration of objects with one object on top of another, connected together by a revolute joint and a rigid joint respectively as shown in Fig. 6.9. The robot used push action to distinguish an articulated object from the rigid object by tracking the motion of the two objects upon performing the action (see Fig. 6.9a,c for articulated object and Fig. 6.9b,d for rigid object).

While pushing actions are sufficient to detect articulated objects with revolute joints, they cannot be used to distinguish objects with prismatic joints. Both robots were used for performing hold-pull manoeuvres for such prismatic objects. In a similar experiment to the revolute joint, two sets of objects were used, a `cube` and `cuboid` which were joint together with a prismatic joint and a rigid joint respectively. The Panda robot autonomously detects and performs a hold manoeuvre on one of the object and the UR5 robot detects a possible pulling position by grasping the other object and pulling it for a fixed distance of 5 cm in the direction parallel to the major eigenvector of the object. The Panda robot used the force-torque sensor embedded at joint 7 to measure the constant holding force of around 8N which is sufficient to immobilise the object without damaging it. Similarly, the tactile sensors on the inside of the gripper finger pads were used to detect contact while grasping and pulling. The ArtReg tracker was used to detect the change in pose of the objects which was used to distinguish the articulated object from the rigid object with prismatic joint as seen in Fig. 6.8. Five repeated trials were performed for each case: revolute joint, top-revolute joint and prismatic joint with the corresponding object with rigid joints for comparison.

6.3.3 Closed-loop control for Goal-driven Manipulation

Since the closed-loop control required accurate pose tracking, extensive evaluations were performed on various configurations of articulated objects with revolute and prismatic joints as well as for single objects. The target pose was chosen by a human user by placing

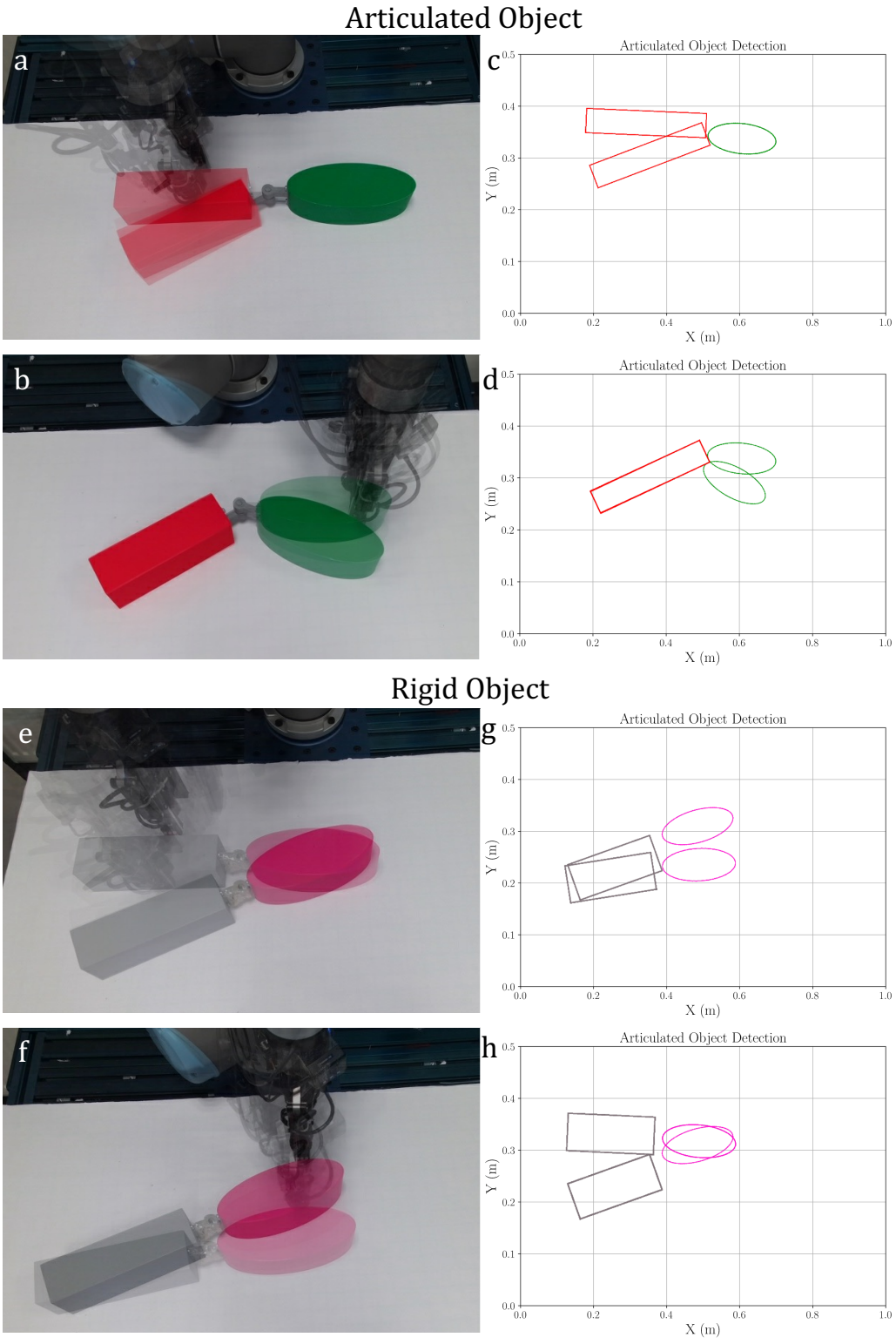


Figure 6.7: Detection of articulated revolute object with interactive perception

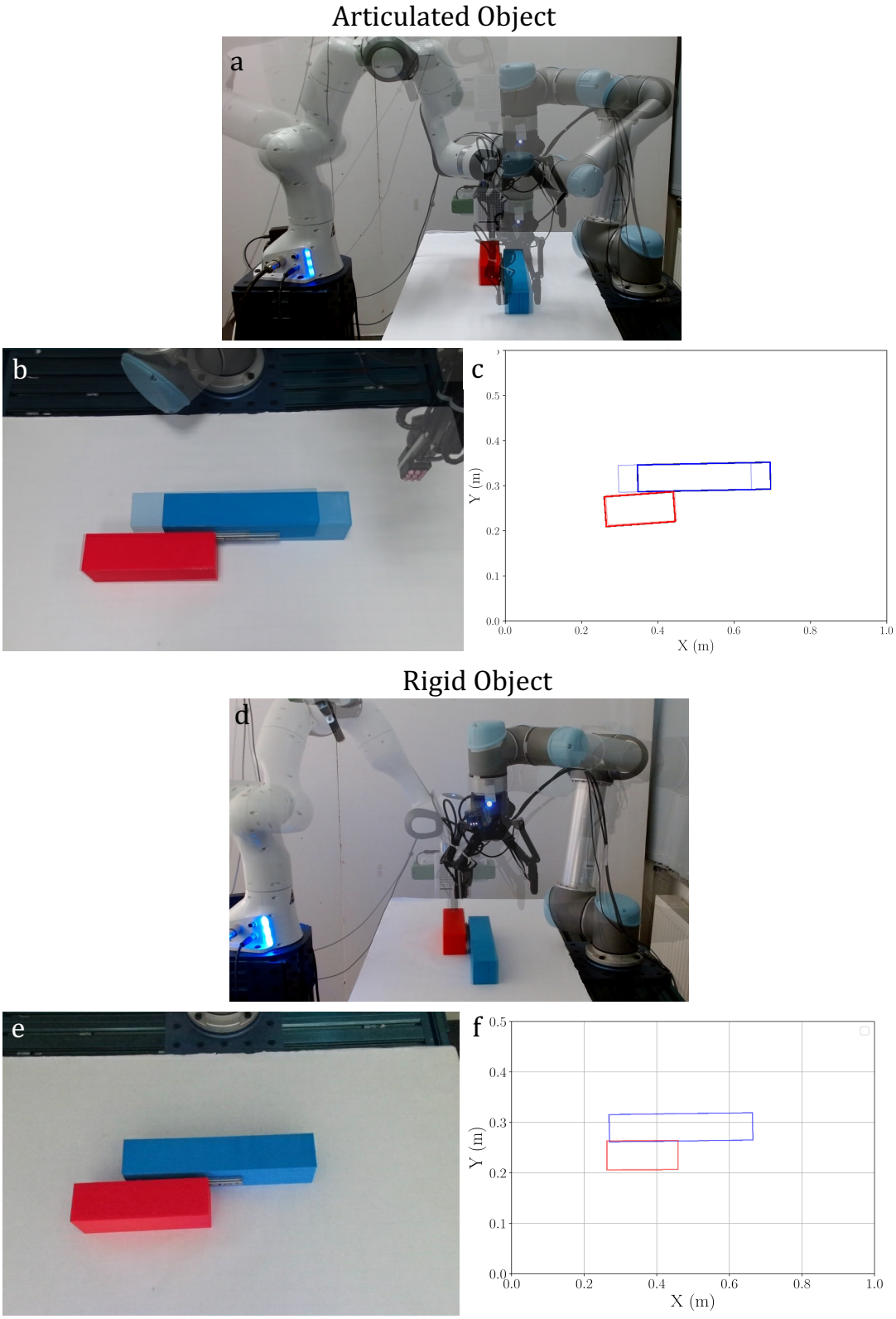


Figure 6.8: Detection of articulated prismatic object with interactive perception

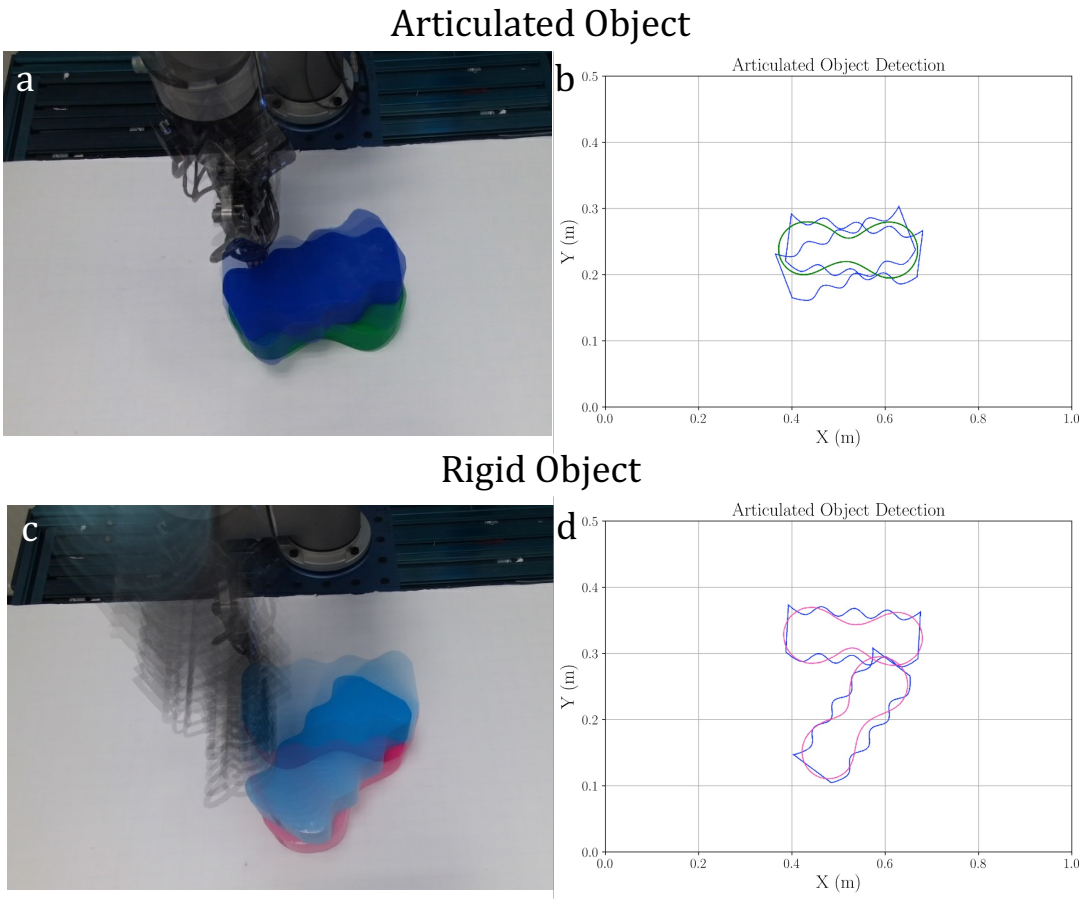


Figure 6.9: Detection of articulated object with overlapping revolute joint using interactive perception

the object in desired configuration at any arbitrary location in the robot workspace. A single RGB-D image was captured at the goal position and the part segmentation was performed to record the number of parts to track and the pose was recorded as the target pose. The object was then moved into an arbitrary initial pose in the robot workspace by the human user and the robot was tasked to manipulate the object into the goal-pose configuration. For single as well as articulated objects with revolute joints, the robot relied upon non-prehensile pushing manipulation for moving the object into a desired goal state. The accuracy of the goal-driven manipulation for an object containing N parts is computed by the L^2 norm in the X-Y plane with the final manipulated configuration of each part of the object with the corresponding goal pose as:

$$||e||_2 = \sqrt{(x-x_g)_i^2 + (y-y_g)_i^2} \quad for\ i = 1,2,\ldots,N \tag{6.18}$$

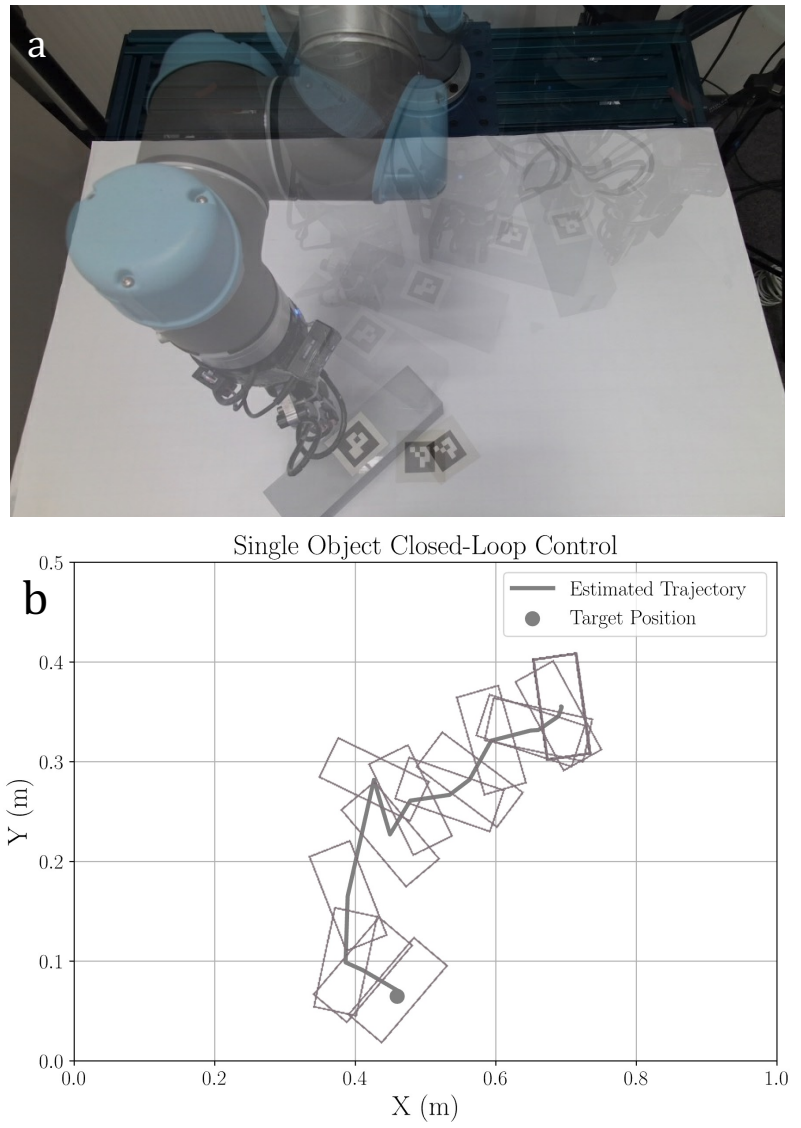


Figure 6.10: Goal-driven closed loop control of single object. (a) Figure shows the robot pushing the object to the goal pose. (b) Figure shows the plot of the estimated trajectory through ArtReg. The dot • represents the goal pose. The rectangles show the poses of the object at discrete time steps.

Consequently, for a single rigid object $N = 1$ in the Eq. (6.18). For single and articulated (revolute) objects, the robot performed iterative incremental pushing of each part one-by-one such that the distance to goal was minimised. This can be seen in Fig. 6.10 for single objects and Fig. 6.11 for articulated objects with revolute joints. The closed loop controller based on the proposed ArtReg as described in Sec. 6.2.4 was used for both single and articulated revolute objects. The only exception for articulated revolute object, the controller was designed to minimise the distance to goal for each part of the object equally, hence the robot iteratively pushes the parts of the objects in an alternating manner. Numerical results

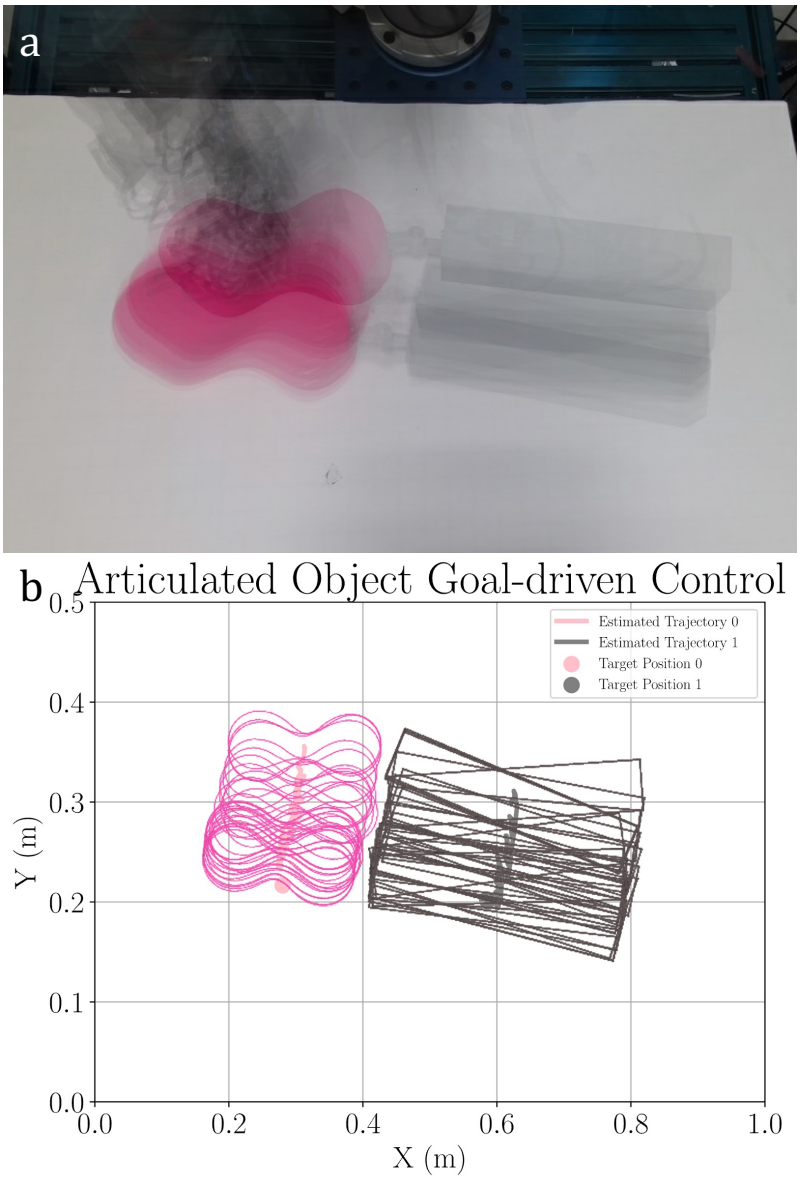


Figure 6.11: Goal-driven closed loop control of articulated object. (a) Figure shows the robot pushing the object to the goal pose. (b) Figure shows the plot of the estimated trajectory through ArtReg. The dots • for each colour represents the goal poses. The object outlines show the poses of the objects at discrete time steps.

are shown in Tab. 6.1. The average error at goal-state for all objects (single/ articulated) was < 3 cm. Five repeated trials were performed for each of the four single objects and all the revolute articulated objects (except `blue-sine/green-butter`) shown in Fig. 6.6, resulting in total 55 repeated trials for this experiment.

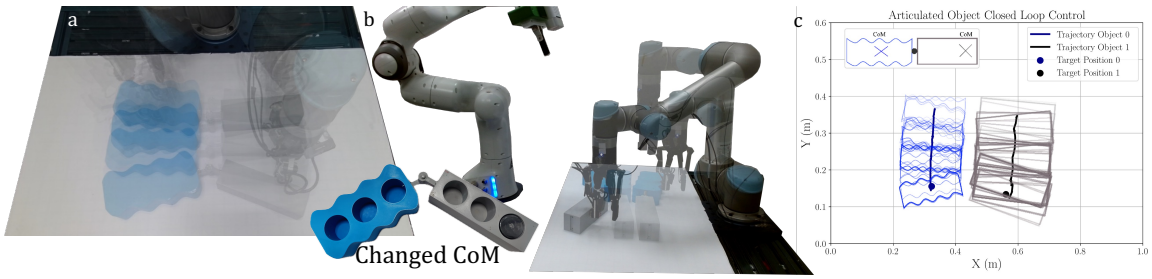


Figure 6.12: Goal-driven Pushing with varied Center of Mass: Articulated Object

Goal-driven Manipulation for Prismatic Articulated Objects

For articulated objects with prismatic joints, the desired pose is specified by the user by translating one segment of the articulated object to an arbitrary distance along the axis of articulation. Once the goal-pose of each part is extracted by using the part segmentation method, the user moves the parts back to an arbitrary initial pose along the axis of articulation. The task of the robot is to identify and track the current pose and manipulate the object into the goal pose. The hold-pull manipulation manoeuvrer is used as it is ideal for manipulating such prismatic articulated objects. The Panda robot which is equipped with the RGB-D camera on its end-effector (in an *eye-in-hand* configuration) performs the hold manoeuvrer and the UR5 robot performs the grasp and pull manoeuvrer. While the Panda robot is performing the holding manoeuvrer, the object is very close to the camera and object tracking stops. Hence, for the grasp and pull manoeuvrer, the UR5 robot relies upon the grasp pose detected prior to interaction. However, the tactile sensors allows the UR5 robot to adapt to minor grasp pose error. The gripper opens to the maximum limit and closes gradually until the tactile sensors detects contact. The UR5 robot performs a pull manoeuvrer along the axis of articulation to the goal-pose. The pulling distance is calculated as the difference between goal pose and current pose. During the pull manoeuvrer, the 3-axis contact force is monitored from each taxel in the tactile sensor array to detect possible loss of contact. Furthermore, a constant force of 5N is exerted by the gripper on the object. The grasping force is sufficient to overcome friction in the prismatic joint. Ten repeated trials were executed for each of the two prismatic objects depicted in Fig. 6.6, totalling 20 repeated trials. The average error of the prismatic articulated objects (1.03 ± 0.96 cm) is less than that of revolute articulated objects (2.30 ± 0.71 cm) due to limited displacement from the initial configuration possible with the prismatic joint.

Table 6.1: Goal-driven Closed Loop Control

Experimental Condition	Object Type	Euclidean Error (cm)
Standard	Articulated Revolute Object	2.302 ± 0.71
	Articulated Prismatic Object	1.031 ± 0.96
	Single Object	1.908 ± 0.86
Varied Center-of-mass	Articulated Object	3.570 ± 1.04
	Single Object	3.604 ± 0.43
Challenging Background	Articulated Object	3.816 ± 0.97
	Single Object	3.101 ± 0.24
Low light	Articulated Object	3.677 ± 0.43
	Single Object	2.974 ± 0.92

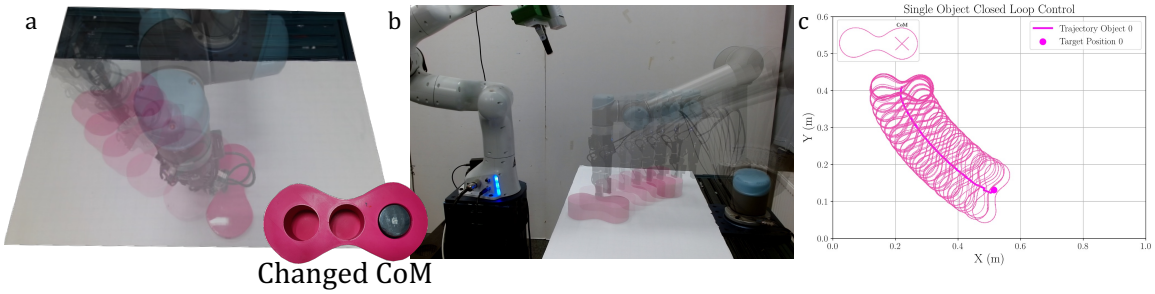


Figure 6.13: Goal-driven Pushing with varied Center of Mass: Single Object

Goal-driven Pushing with Varied Centre of Mass (CoM)

Typically it is assumed that the CoM of an object coincides with the geometric centroid which can be computed as the mean of all the 3D points representing the object. However, in many scenarios this assumption fails to hold true. Hence, an experiment was designed in which an additional weight of 500g in the form of a calibrated weight metal cylinder was inserted into the object. The objects were 3D printed with holes embedded within which are capable of housing external objects. Depending on the object shape, the weight of individual objects vary between 200g-400g. Hence, the additional weight shifts the CoM of the object from the geometric centroid. However, through visual inspection alone the object with varied CoM was indistinguishable from an identical shaped object without the additional weight inserted into it. This experiment evaluated the ability of the approach to push the object(s) to the goal configuration with varied CoM. The shifted CoM caused the object to turn when being pushed through the geometric centre. While inferring the CoM position through interaction goes beyond the scope of this work, the robot relied

upon visual and tactile feedback to compensate for the undesired motions caused during pushing. This is shown in Fig. 6.12 for pushing articulated revolute object wherein the CoM of the grey `cuboid` object was shifted with 500g placed on the right hole. Five repeated experiments were performed for the external weight placed in each of the three holes for the `pink-butter` single object and three repeated trials for each of the six holes for `blue-sine/gray-cuboid` articulated object resulting in a total of 33 repeated trials. As seen from Fig. 6.12c, the robot was capable of pushing the articulated object to the goal state with an error of 3.816 ± 0.97 cm. A similar result was also seen with a singular object with shifted CoM while pushing to the goal pose as in Fig. 6.13c with the average error in the goal state being 3.604 ± 0.43 cm.

Goal-driven Pushing with Challenging Background

As the part-segmentation approach relied upon visual point clouds and separating the foreground (object) from background points, an experiment was performed with a challenging coloured pattern background with the articulated and single objects for goal-driven pushing as seen in Fig. 6.14 and Fig. 6.15. To increase the complexity, a revolute articulated object with all parts having the same colour was used. Although the part segmentation approach was based on RGB point cloud data for the region growing method, it was demonstrated that it is unaffected by challenging coloured backgrounds as seen by the object trajectory and proximity to the target pose in Fig. 6.14b, Fig 6.15b. A total of 20 repeated for single and articulated objects respectively were performed. The average error for target-driven control was 3.816 ± 0.97 cm for articulated objects whereas it was 3.1 ± 0.24 cm for single objects with challenging backgrounds. The discrepancy in error was approximately 1.5 cm greater, on average, in comparison to corresponding objects presented against a standard white coloured background. It can be noted that the challenging background colour does not adversely affect the ArtReg tracking algorithm as well as closed-loop control approach.

Goal-driven Pushing in Low-Light Conditions

Furthermore for robustness testing, goal-driven pushing experiments with single objects and articulated objects were also performed in low light conditions. It is well known that vision-based methods are susceptible to diminished lighting conditions. During the experiments, the overhead lights in the room were switched off and minimal light was emanating from the computer screen that was used to control the robot as seen in Fig. 6.16a. While the part segmentation method relied upon visual point clouds, the robot interaction relied upon tactile sensing as well. In instances where an erroneous pushing configuration was identi-

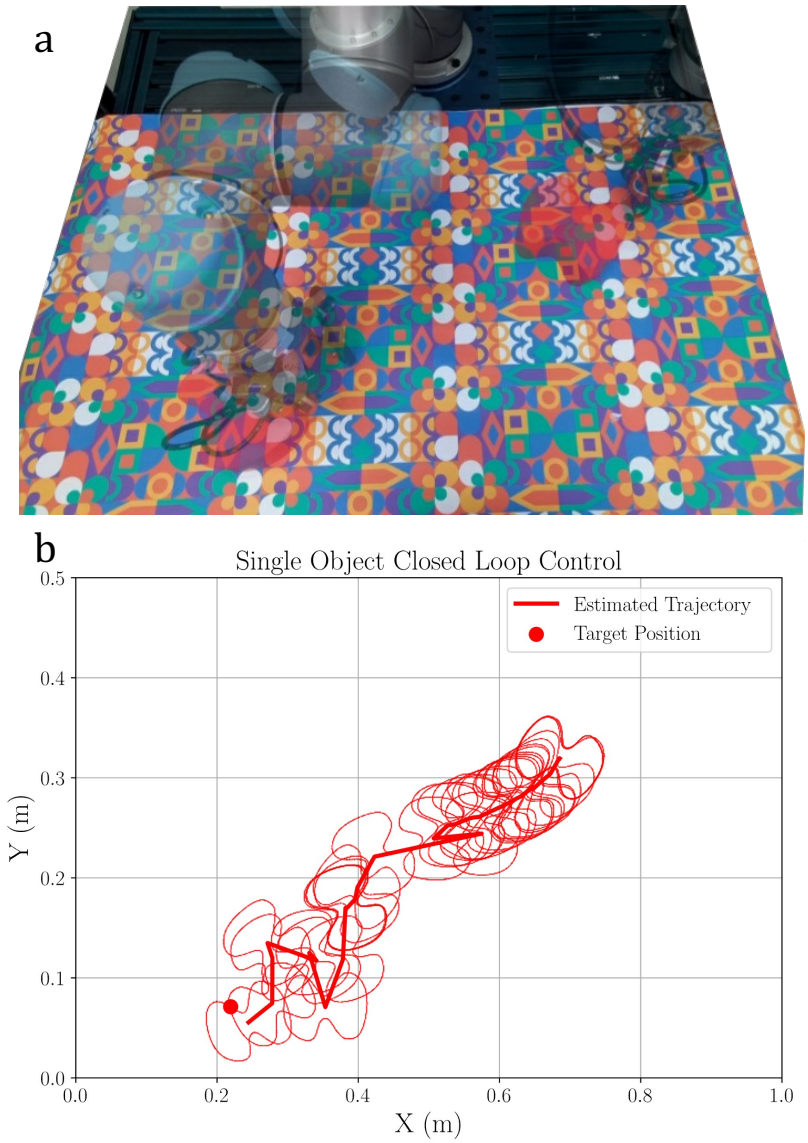


Figure 6.14: Goal-driven Pushing with challenging background: Single Object

fied, the robot upon detecting unintended contact during movement towards the designated pose, promptly halts its operation and reverts to its initial position. By leveraging vision and tactile sensing the robot was able to push the objects to goal within the prescribed margin of error as seen the trajectory shown in Fig. 6.16b. A total of 20 repeated for single and articulated objects respectively were performed. The average error in low light conditions for articulated objects was 3.67 ± 0.43 cm and for single objects was 2.97 ± 0.92 cm.

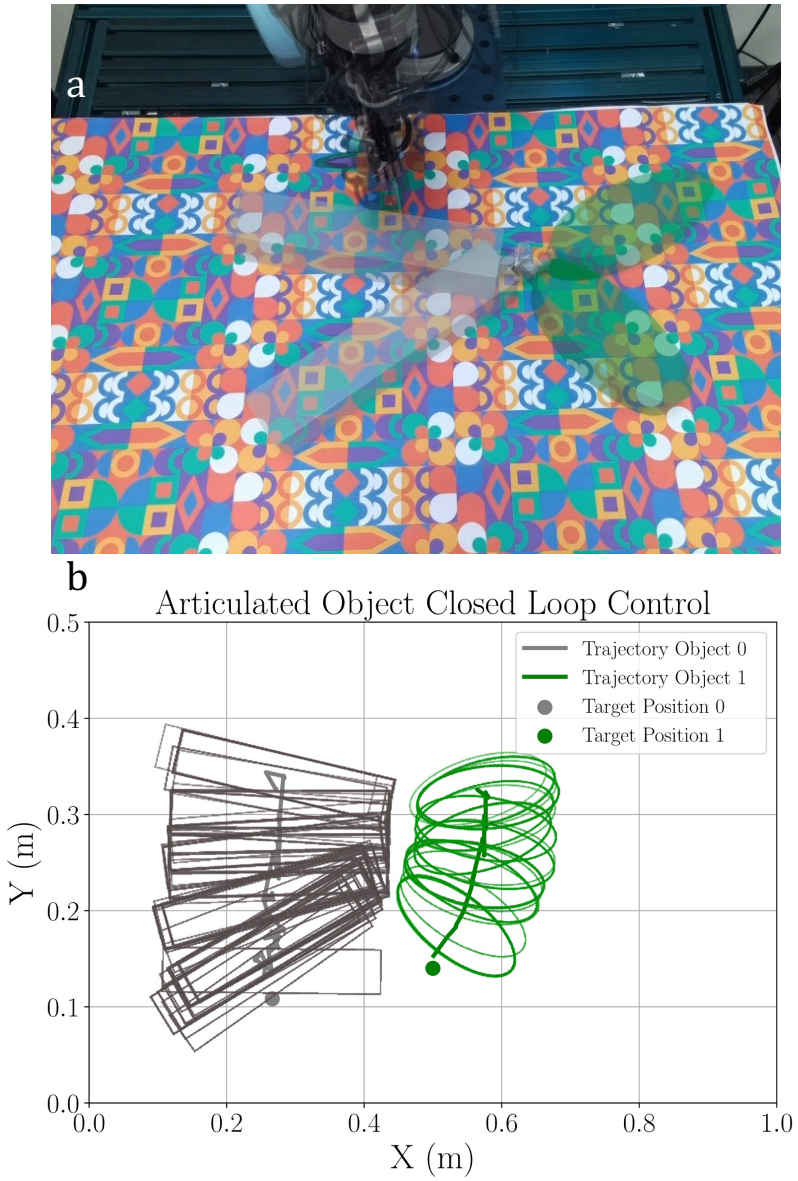


Figure 6.15: Goal-driven Pushing with challenging background: Articulated Object

6.3.4 Object Tracking

To isolate and precisely assess the accuracy of the ArtReg tracker without any error propagation stemming from the closed-loop controller, the robot was manually operated via the teach pendant to manoeuvre the object through a series of arbitrary poses. Furthermore, additional experiments were also performed wherein multiple objects were moved in the workspace by the user in randomised trajectories. Aruco markers were placed on each object which were used to gather the ground-truth poses. The average euclidean distance error calculated at 5 frames-per-second from Eq. (6.18) is presented in Tab. 6.2. In total,

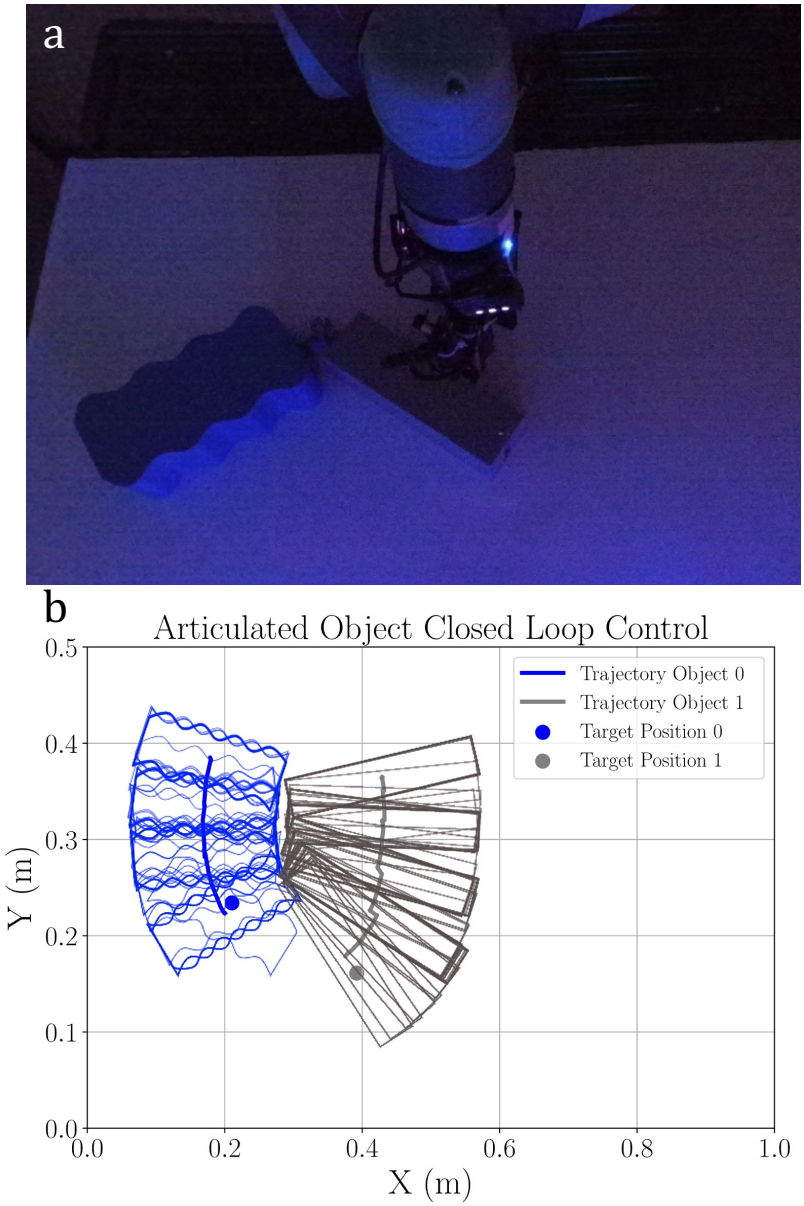


Figure 6.16: Goal-driven Pushing with low ambient light conditions: Articulated Object

10 repeated trails were performed for each type of tracking experiment: single, multiple and articulated. For all objects (single, multiple and articulated), the average pose tracking error with the proposed ArtReg algorithm was less than 2 cm. The least tracking error was achieved for single objects as expected with an average error of 1.31 cm whereas the tracking error increases marginally for articulated objects (wherein 2 objects are tracked) with an average error of 1.41 cm and for multiple objects (with the object number N ranging from $3 \leq N \leq 7$) with an average tracking error of 1.51 cm.

Table 6.2: Object Tracking

Object Type	Euclidean Error (cm)
Single Objects	1.313 ± 0.27
Multiple Objects	1.512 ± 0.31
Articulated Objects	1.406 ± 0.48

6.3.5 Baseline Comparison

The ArtReg pose tracking algorithm was compared with the following state-of-the-art methods: (a) particle filter-based tracker that has been used in many prior works (Cabido et al., 2009; Gonzales and Dubuisson, 2015; Cifuentes et al., 2016); (b) learning-based approach termed ANSCH in (Li et al., 2020b) and (c) FilterReg algorithm presented in (Gao and Tedrake, 2019). A standard benchmark dataset was used for comparison termed as the PartNet-Mobility dataset (Xiang et al., 2020). The PartNet-Mobility dataset consists of various real-world articulated mesh models with movable part definitions. The models from the following categories are chosen: dishwasher (1 DoF revolute joint), glasses (two 1 DoF revolute joints), drawer (1 DoF prismatic joint), and blade (1 DoF prismatic joint). The chosen articulated objects are shown in Tab. 6.3. The object models were provided as URDF files which were simulated in a PyBullet simulator (Coumans and Bai, 2016). For all the objects, the joint values were in the range $[0, 1]$. Ten different poses were randomly sampled for each object by choosing different values for the joint angle/ distance in the range $[0, 1]$ and articulating the moving link according to the joint value, in total 40 experimental trails. For each pose, a point cloud was captured by a simulated depth camera at a 45° top-down viewing angle. This results in partial point clouds for the objects as seen from Tab. 6.3 and provides a challenging benchmark for all the baseline methods as well as the proposed approach. As the focus was on the tracking accuracy, the model point clouds of each link were provided as input which were sampled from the CAD mesh files.
Baseline Implementation Details: The implementation of the FilterReg algorithm (Gao and Tedrake, 2019) available in the Python package `probreg` (Kenta-Tanaka, 2019), ANSCH method from official GitHub implementation (Li et al., 2020b) and particle filter reimplemented using Open3D modules (Zhou et al., 2018b) for point cloud processing were used. All baseline approaches were implemented in Python. For quantitative evaluation, the Average Distance of model points with Indistinguishable views metric (ADI) was used which is insensitive to object symmetries (Hinterstoisser et al., 2013). The ADI metric is

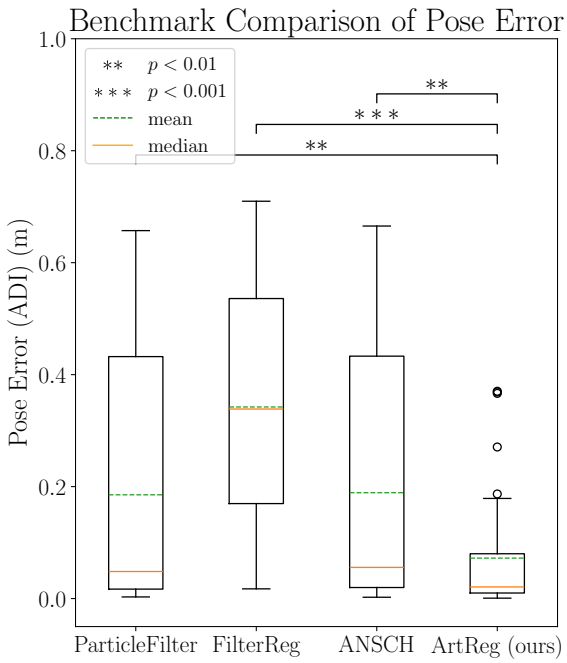


Figure 6.17: Pose estimation results with simulated articulated objects from PartNet-mobility dataset in random pose configurations with comparisons against state-of-the-art. p values calculated by Welch's t-test shown as ∗.

measured as follows :

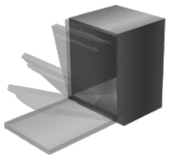
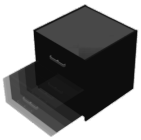
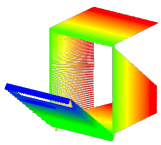
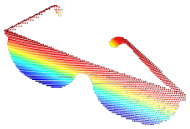
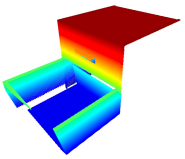
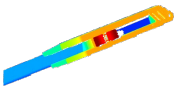
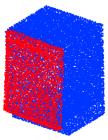
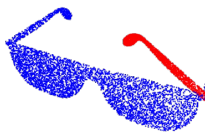
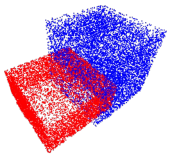
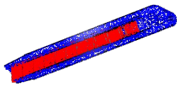
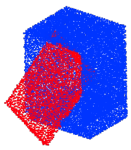
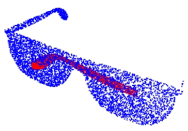
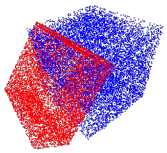
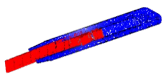
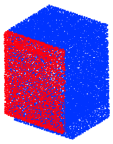
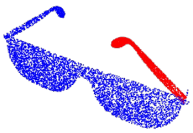
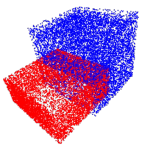
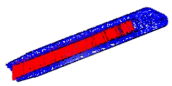
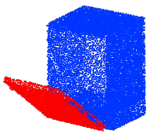
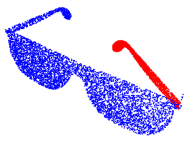
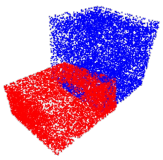
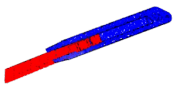
$$\texttt{err}_{adi} = \frac{1}{|\mathcal{O}|} \sum_{\mathbf{p}_1 \in \mathcal{O}} \min_{\mathbf{p}_2 \in \mathcal{O}} ||(\mathbf{R}_{gt}\mathbf{p}_1 + \mathbf{t}_{gt}) - (\mathbf{R}_{est}\mathbf{p}_2 + \mathbf{t}_{est})|| \tag{6.19}$$

where $(\mathbf{R}_{gt}, \mathbf{t}_{gt})$ and $(\mathbf{R}_{est}, \mathbf{t}_{est})$ refers to ground-truth and estimated rotation and translation respectively, $\mathcal{O}$ refers to the object model point cloud and the points $p_1 \in \mathcal{O}$ and $p_2 \in \mathcal{O}$ belong to the object point cloud and denote the closest corresponding points when $\mathcal{O}$ is transformed by $\{\mathbf{R}_{gt}, \mathbf{t}_{gt}\}$ and $\{\mathbf{R}_{est}, \mathbf{t}_{est}\}$ respectively. The quantitative comparison against baseline approaches is presented in Fig. 6.17. Furthermore, object-wise results are presented in Fig. 6.18. Qualitative pose estimation results for selected poses are presented in Tab. 6.3 and discussed in the following section.

6.4 Discussion

In this chapter, a full-fledged framework for articulated object detection, pose estimation, and tracking as well as goal-driven closed-loop control has been presented. The proposed ArtReg algorithm has been demonstrated to perform accurate and robust pose tracking of single, multiple, and articulated objects. As illustrated in Fig. 6.17, it is evident that the ArtReg algorithm outperforms the state-of-the-art methodologies: ArtReg

Table 6.3: Simulated articulated objects from PartNet-Mobility dataset (Xiang et al., 2020) used for benchmarking the proposed ArtReg method against state-of-the-art algorithms.

	Dishwasher	**Glasses**	**Drawer**	**Blade**
Object				
Articulation Type	1 DoF Revolute Joint	2 DoF Revolute Joint	1 DoF Prismatic Joint	1 DoF Prismatic Joint
Measured point cloud				
Particle Filter				
FilterReg (Gao and Tedrake, 2019)				
ANSCH (Li et al., 2020b)				
ArtReg (ours)				

demonstrated an average accuracy improvement of approximately 60% and a reduction in median error exceeding 50% compared to baseline methods : FilterReg (Gao and Tedrake, 2019), particle filter (Cabido et al., 2009; Gonzales and Dubuisson, 2015; Cifuentes et al., 2016) and ANSCH (Li et al., 2020b) ($p < 0.01$ as determined by Welsh's t-test). The mean ADI error value for the ArtReg algorithm was 7.23 cm and median ADI error was 2.1 cm. Whereas for the state-of-the-art algorithms, the average ADI errors were comparatively larger: particle filter: 18.53 cm, FilterReg (Gao and Tedrake, 2019): 34.22 cm and ANSCH (Li et al., 2020b): 18.91 cm. The object-wise pose results in Fig. 6.18 demonstrates that ArtReg provided highly accurate estimates for articulated objects with completely measured point clouds without self-occlusions (such as glasses) with average ADI

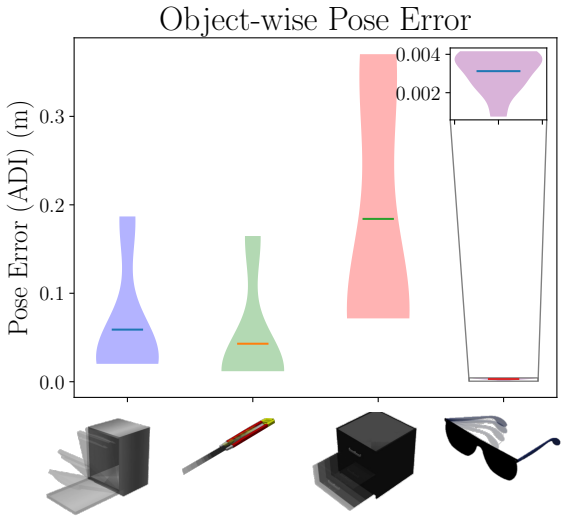


Figure 6.18: Violin-plots showing the object-wise pose estimation results with the ArtReg method. The notches in the violin-plot shows the mean value.

error of 0.312 cm and relatively larger errors for articulated objects such as drawer (average ADI error 18.4 cm) due to high self-occlusion as well as the larger size (bounding box size upto 2 m). The superiority of the proposed methodology over state-of-the-art techniques was qualitatively evident in Tab. 6.3. Specifically, existing methods such as (Gao and Tedrake, 2019) and (Li et al., 2020b) demonstrated a deficiency in accurately capturing the pose of articulated objects, such as dishwashers or drawers, whereas ArtReg method yields precise estimations. Furthermore, the ArtReg approach does not require any prior knowledge of the object whereas the recent state-of-the-art approaches such as ANSCH (Li et al., 2020b) required prior training on a large labelled dataset of category-level object with articulation and pose information.

Considering real-world robotic experiments, the proposed framework achieves highly accurate pose tracking ($<$ 1.5 cm average error over the trajectory) and goal-driven closed-loop control ($<$ 4 cm average error at goal-state). The robustness of the proposed approach has been demonstrated in various conditions such as low ambient light, challenging backgrounds, and varying the centre-of-mass of the objects. The importance of tactile perception in the framework is noted: for the detection of articulated objects, where objects have identical visual features (such as Fig. 6.8), robots relied on tactile feedback during interactive perception to discern the articulated object. During goal-driven manipulation, the UR5 robot relied on tactile force feedback to ensure contact during manipulation, and in case of loss of contact, the framework re-triggers the computation of the push or hold-pull pose to perform the manipulation task until the goal-state was achieved. Furthermore, tactile feedback was crucial in situations such as low-light conditions, and when the CoM of the

object was varied by the user. Although the CoM was not directly inferred, the tactile feedback during the manipulation manoeuvres and the resulting visual feedback of the effect of the manipulation allowed the robot to adapt the pushing strategy such that the goal-pose was achieved. The synergistic combination of visual and tactile perception allowed the two robots equipped with the complementary sensing abilities to detect, track and manipulate various types of objects in a robust manner.

However, there were limitations in the proposed framework which can be considered as part of future work: the pose tracking method relied upon an accurate point cloud segmentation to provide various parts of an articulated object. While an off-the-shelf region-and-colour-based segmentation algorithm has been used, for more complex scenes, recent large vision foundation models such as Segment Anything (Kirillov et al., 2023) may be used as a drop-in replacement in the framework. Goal-driven manipulation has been demonstrated on 3D-printed objects of relatively simpler primitive shapes. By 3D printing, the objects could be modified easily to test for variations such as off-symmetric centre-of-mass. As the main contribution of this chapter is the pose tracking algorithm namely ArtReg, the evaluation of the accuracy has been performed on these 3D-printed objects as well as in simulation with complex real-world objects such as glasses, dishwashers and drawers. However, for an extension of the goal-driven manipulation to more complex and real-world 3D objects such as dishwashers and drawers, complex motion planning and control strategies are necessary. Recent works have used tele-operated setups with behaviour cloning to enable robots to manipulate complex articulated objects which can be a promising direction of future work (Xiong et al., 2024; Eisner et al., 2022). Hardware improvements such as robust multi-fingered hands with tactile sensing may be necessary for complex manipulation tasks which provides increased dexterity compared to antipodal grasps with 2 finger grippers.

To summarise, in this chapter, a novel SE(3) Lie Group-based Unscented Kalman Filter approach for real-time object tracking termed ArtReg has been presented. The proposed approach was demonstrated to perform robustly and accurately with various types of objects such as single, multiple and articulated objects under different conditions such as low-light, challenging backgrounds and varying centre-of-mass. Visual and tactile perception were seamlessly integrated in the full-fledged framework which allowed the robots to detect any possible kinematic articulation in objects using interactive manipulation, track the object using the ArtReg tracker and perform closed-loop goal-driven manipulation to bring the objects to the desired goal-state. The two-robot team equipped with visual and tactile sensing respectively performed various types of manipulation manoeuvres such as pushing or hold-pulling depending on the type of articulated object (revolute or prismatic joints) to execute goal-driven manipulation. The ArtReg algorithm was also benchmarked against

various state-of-the-art approaches and a clear outperformance was shown on a standard benchmark dataset consisting of various types of articulated objects.

In the context of practical applications, the ArtReg algorithm discussed in this chapter demonstrates its utility in real-time tracking of the pose of articulated objects, with potential applications in household settings, human-robot collaborative manufacturing, and geriatric care (Katz and Brock, 2008). Furthermore, the methodology presented for detecting articulated joints through interactive visuo-tactile manipulation using pushing and bi-manual hold-pull actions can be applied in scenarios where robots need to interact with fragile and unknown objects, typical in household environments. Evidently, manipulating articulated objects such as eyeglasses, door knobs, scissors, and so on require robust multi-finger robotic hands with distributed tactile sensing. Although advanced robotic hardware is increasingly becoming commercially and economically feasible (Piazza et al., 2019), the ArtReg algorithm remains hardware-agnostic and can be applied to evolving technological advancements. This capability is due to its reliance on raw 3D point clouds, which are accessible through both visual sensors and tactile sensors integrated within robotic hands.

Part III

Cross-Modal Visuo-Tactile Perception for Object Recognition

Chapter 7

Visuo-Tactile Cross-Modal Perception for Object Recognition

Parts of this chapter are **published** as:

- "*Deep Active Cross-Modal Visuo-Tactile Transfer Learning for Robotic Object Recognition*", **P. K. Murali**, C. Wang, D. Lee, R. Dahiya, and M. Kaboli, in IEEE Robotics and Automation Letters, 2022 Jul 15;7(4):9557-64 (Murali et al., 2022e).
- "*Towards Robust 3D Object Recognition with Dense-to-Sparse Deep Domain Adaptation*," **P.K. Murali**, C. Wang, R. Dahiya, and M. Kaboli, in The IEEE International Conference on Flexible and Printable Sensors and Systems (FLEPS 2022), pp. 1–4 (Murali et al., 2022d).

The video of the experiments from this chapter is available here:
`https://doi.org/10.1109/LRA.2022.3191408/mm1`video-link

7.1 Introduction

Humans from infants to adults can seamlessly transfer the knowledge gained from visual modality to the tactile modality in order to perceive and interact with objects in the environment especially during lack of visual feedback (Martino and Marks, 2000; Sann and Streri, 2007). For instance, we can identify and distinguish previously *seen* objects blindly only through *touch*. The human sensing and perception systems are also active such that the sensory systems are purposefully controlled to increase the information gained for the task at hand (Prescott et al., 2011). This chapter aims to develop an approach to provide similar

abilities to autonomous robots for cross-modal object recognition by actively training with the visual modality and transferring to the tactile modality without explicit training with the tactile modality as shown in Fig. 7.1. This can provide increased autonomy and resilience for robots in unstructured environments. If visual sensing is unavailable due to various reasons such as occlusions, limited field of view, change in light intensity, dust blocking the sensor and so on, the robot is capable of completing the object recognition task using the tactile modality by leveraging only the previously gained knowledge from vision (Li et al., 2020a). Furthermore, training an object recognition model with tactile sensing is time consuming due to sparsity of tactile data, human annotation and need for interaction with objects whereas through cross-modal learning, the robot can exploit the *a priori* gained knowledge using visual sensing to recognise objects during *testing* stage through only tactile sensing. Moreover, through active tactile perception and learning, the robot can autonomously reduce the number of actions to perceive objects physical properties and to learn efficiently about objects and discriminate them among each other (Kaboli et al., 2018). As detailed in Chap. 2, prior works in visuo-tactile cross-modal recognition have certain limitations such as they equalize the point density between the visual and tactile point clouds to ease the challenge of cross-modal transfer (Falco et al., 2019). Due to the extreme variation of point density (as seen from Fig. 7.1), this leads to loss of information in the visual data. Furthermore, previous works in this domain rely upon pre-recorded visuo-tactile data that are typically collected through manual teleoperation of a robot. Leveraging active perception and learning techniques can aid in reducing data collection costs and improve time efficiency for vision-to-tactile cross-modal domain adaptation.

The contributions of this chapter are as follows:

(I) A novel framework for deep active visuo-tactile cross-modal robotic object recognition. The deep neural network (*xAVTNet*) is trained with dense point cloud data from visual sensors and tested on sparse point clouds acquired from tactile sensors.

(II) A novel unsupervised domain adaptation loss function termed *VTLoss* for minimising the domain gap between the visual and tactile domain.

(III) An *active* deep learning framework for visual object learning for reducing redundant data collection and an *active* tactile-based object recognition approach by leveraging *xAVTNet* to recognise objects online with minimal number of touches.

Extensive experiments were performed to show the validity of the proposed approach against baseline and state-of-the-art approaches with real robot experiments.

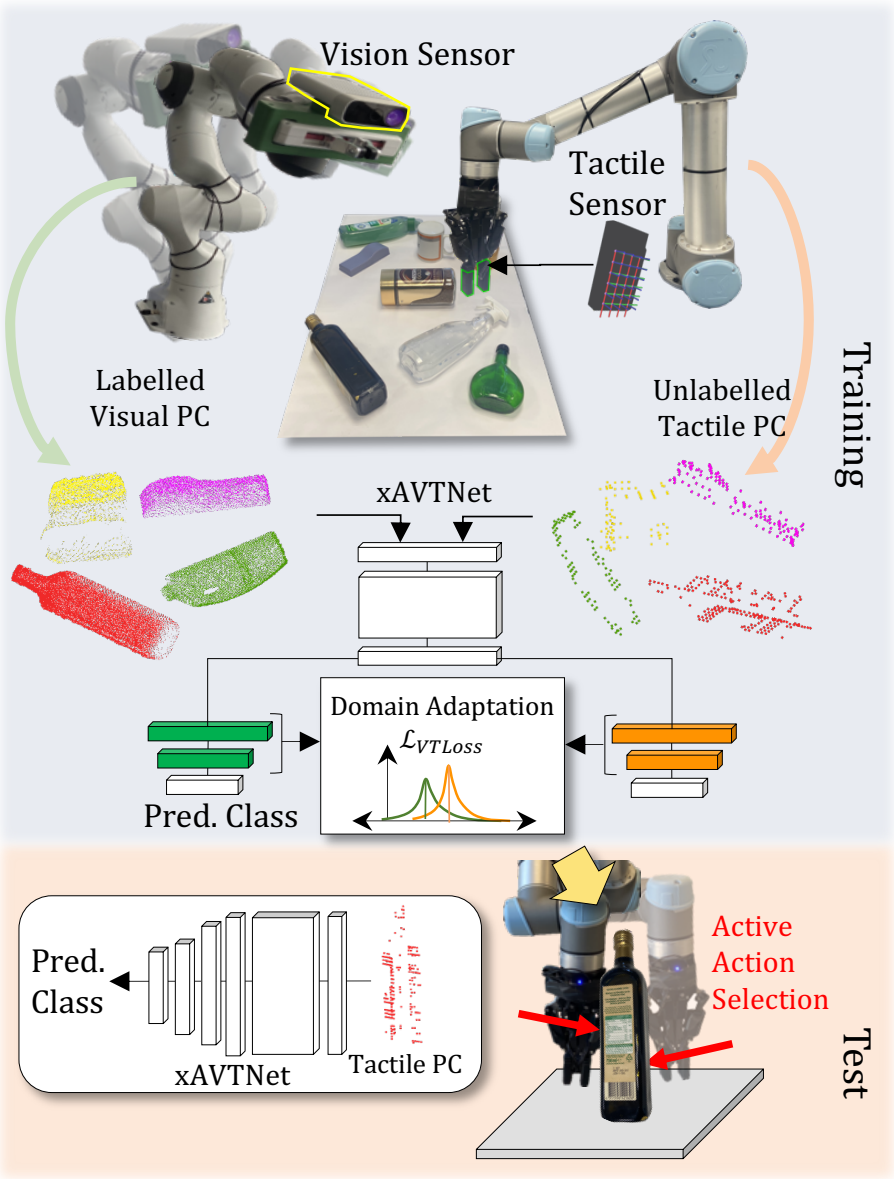


Figure 7.1: Experimental setup: A Franka Emika Panda robot with a RGB-D vision sensor on the end-effector for active visual perception and learning. A UR5 robot with 3-axis tactile sensors on the gripper for deep cross-modal visuo-tactile transfer learning and active tactile object recognition.

7.2 Methodology

7.2.1 Problem Description

A novel framework shown in Fig. 7.2 is proposed herein for the task of deep active visuo-tactile cross-modal object recognition. The network *xAVTNet* is trained with labelled source domain dataset $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ with n_s samples from vision domain constructed

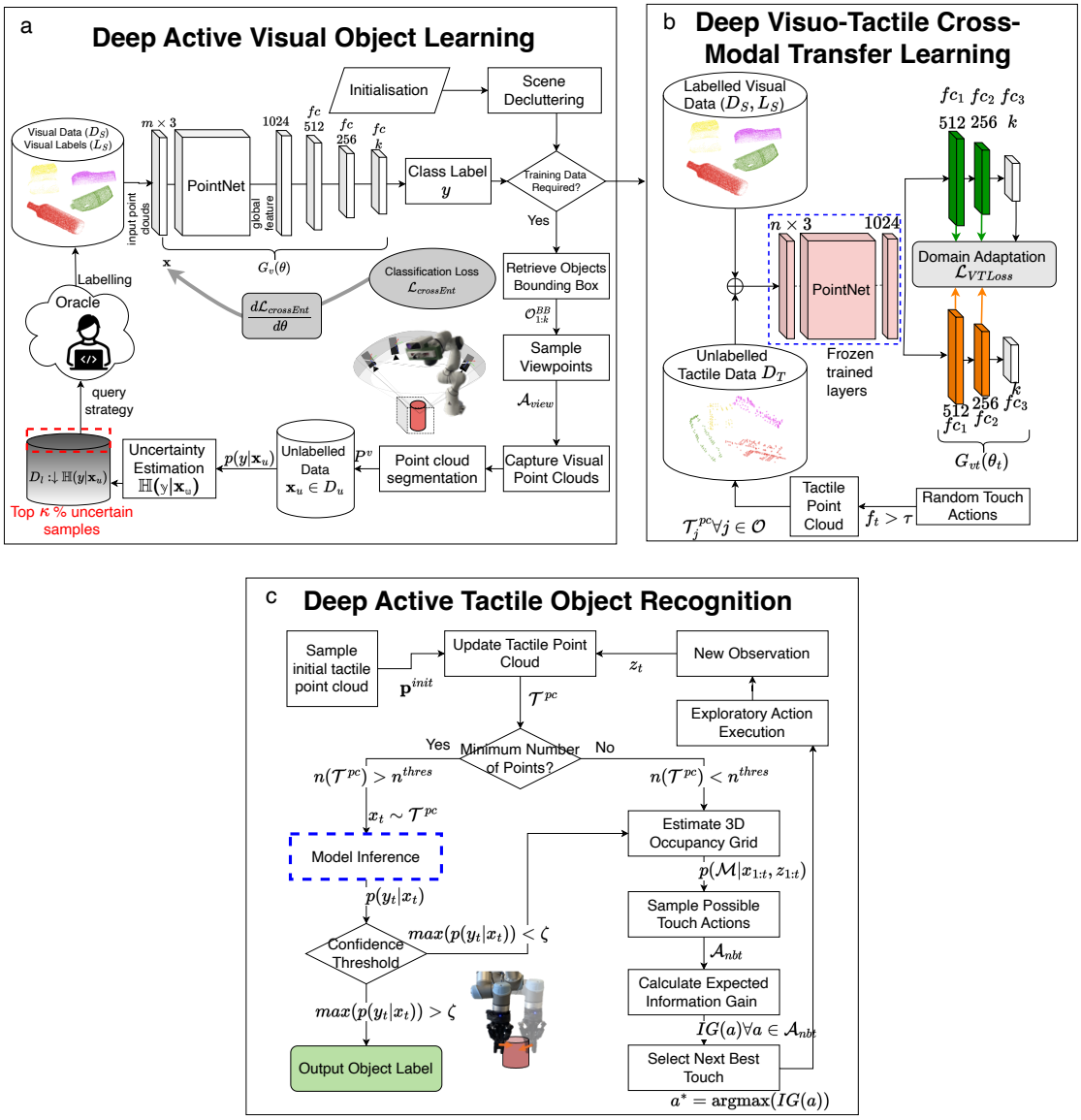


Figure 7.2: Proposed framework for deep active visuo-tactile cross modal object recognition

using an active learning strategy by querying uncertain samples from a larger unlabelled dataset D_u consisting of n_u samples with $n_u \gg n_s$ (Fig. 7.2a). Given the labelled source domain $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ from vision domain and unlabelled target domain $D_t = \{x_j^t\}_{j=1}^{n_t}$ with n_t samples from tactile domain, the model is adapted by reducing the domain discrepancy through the proposed *VTLoss* (Fig. 7.2b). The adapted model is used for active tactile-based object recognition wherein the robot is tasked to reason upon possible tactile touch actions to perform and chooses the next best touch which maximises the expected information gain (Fig. 7.2c).

7.2.2 Deep Active Visual Object Learning

Network Architecture: Point clouds are captured from each object using the vision sensor which are provided as input to the network. The input to the network is of the order $m \times 3$ representing m points having dimensions x, y, z. The network outputs k probabilistic classification scores for all k candidate classes. PointNet (Qi et al., 2017) was used as the backbone for feature extraction. PointNet applies input and feature transformations and aggregates the point features by max pooling to a global feature vector of size 1024 (Qi et al., 2017). The global feature vector is followed by three fully-connected (fc) layers of size $512, 256, k$. The mapping from input point clouds to output classes is denoted as G_v and associated parameters by θ. The *xAVTNet* was trained with dense point clouds from the vision sensor of real world objects with $k = 12$ classes. The visual point clouds were subsampled to 1024 points before passing to *xAVTNet*. The cross-entropy loss was used for training. The visual point cloud dataset D_s representing the source domain was collected using an active learning technique as detailed below.

Visual Viewpoint Sampling and Visual Data Collection: In order to collect visual training data of the objects, a vision sensor/camera was used that was attached to a manipulator robot (Franka Emika Panda) capable of choosing arbitrary viewpoints in 3D space limited by the workspace and kinematic constraints of the robot. Choosing different viewpoints of the same objects helps to improve the predictive accuracy of the network as the same object can appear differently based on the view. For example, two identical mugs, one featuring a handle and the other not, could present substantial difficulties for a network trained on samples wherein the handle is perpetually occluded due to the camera's static viewpoint. While commanding the robot to arbitrary viewpoints, it is crucial to maintain the viewing angle of the camera such that the object of interest lies within its field of view (FoV). A viewpoint $a^{view} \in \mathcal{A}^{view}$ is defined as the 3D position $\mathbf{p}^{view} \in \mathbb{R}^3$ and orientation $\mathbf{R}^{view} \in SO(3)$ of the camera frame. Markov Monte-Carlo sampling of N viewpoints is performed on the hemisphere space located above the centroid $\mathbf{o}_{centroid}$ of the target object which is known *a priori*. The 3D position $\mathbf{p}^{view}$ is randomly sampled as a point on the hemisphere and the orientation of the view as axis of rotation $\vec{\mathbf{e}}$ and angle θ is computed with:

$$\vec{\mathbf{h}} = \frac{\mathbf{p}^{view} - \mathbf{o}_{centroid}}{||\mathbf{p}^{view} - \mathbf{o}_{centroid}||} \tag{7.1}$$

$$\theta = \cos^{-1}(\vec{\mathbf{h}} \cdot \vec{\mathbf{Z}}), \quad \vec{\mathbf{e}} = \frac{\vec{\mathbf{h}} \times \vec{\mathbf{Z}}}{||\vec{\mathbf{h}} \times \vec{\mathbf{Z}}||} \tag{7.2}$$

where $\vec{\mathbf{Z}} = \{0, 0, 1\}$ is the Z-axis of the world frame. Using the resulting angle-axis for-

mulation $(\vec{\mathbf{e}}, \theta)$ or equivalent rotation matrix $\mathbf{R}^{view}$ from Eq.(7.2), the camera is always oriented towards the object of interest. The robot is commanded to N viewpoints sequentially and the point clouds are extracted from each viewpoint. The raw point clouds were cleaned in order to remove outlier noisy points as well as the base plane and added to the unlabelled dataset $\mathbf{x}_u \in D_u$.

Uncertainty Estimation and Query Strategy: The goal of active learning is to select samples from the unlabelled dataset D_u, which upon labelling and training improves the model accuracy significantly with fewer training samples. In order to select such samples from the unlabelled dataset, the predictive probability of the network $p(y|\mathbf{x}_u)$ is used to determine uncertainty. The softmax function provides the predictive probability of an input sample. However, as noted in previous works (Feng et al., 2019; Beluch et al., 2018), the softmax function may provide inconsistent predictions as it provides higher probability to unseen data. Hence, the Monte Carlo dropout (MC-dropout) method is adopted instead to extract the uncertainty (Gal and Ghahramani, 2016). The MC-dropout technique (Gal and Ghahramani, 2016) casts dropout training in deep neural networks as an approximate Bayesian inference in deep Gaussian processes. It works by performing multiple stochastic feed-forward passes through the network with dropout active at test time and averaging the results. In particular, it is defined as

$$p(y|\mathbf{x}_u) = \frac{1}{T}\sum_{t=1}^{T} p(y|\mathbf{x}_u, \mathbf{W}_t) \tag{7.3}$$

where $\mathbf{W}_t$ refers to the weights of the network at the t^{th} inference and T refers to the total number of stochastic forward-passes. Given the predictive probability, the uncertainty of the samples can be quantified by measuring the Shannon Entropy (Shannon, 2001) as:

$$\mathbb{H}(y|\mathbf{x}_u) = -\sum_{c=1}^{C} p(y=c|\mathbf{x}_u) log p(y=c|\mathbf{x}_u) \tag{7.4}$$

where $c = 1, 2, \ldots C$ refers to C classes. The unlabelled dataset was ordered based on the Shannon entropy values and top κ samples were queried as the most informative samples for labelling. The selected samples were labelled by the oracle (human annotator) and added to the training dataset for further training.

7.2.3 Deep Visuo-Tactile Cross-Modal Object Learning

The challenge of domain adaptation arises from the fact that the target domain (tactile modality) has no labelled data, hence fine-tuning the network trained on the source

domain to the target domain directly is impossible. Another challenge stems from the density and sparsity of visual and tactile data, respectively. The dense visual data contains 1024 points (in this case), while the sparse tactile data usually contain 30-80 points. The available labelled source data and the unlabelled target data are used to minimise the distributions of the two domains in the latent space of the fully connected layers. The present study employs discrepancy-based methodologies to extract domain-invariant representations for the purpose of unsupervised domain adaptation. Popular techniques in the literature among discrepancy-based methods include Maximum Mean Discrepancy (MMD) (Borgwardt et al., 2006) and Correlation Alignment (CORAL) (Sun et al., 2016). Given labelled source domain $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ with n_s samples and unlabelled target domain $D_t = \{x_j^t\}_{j=1}^{n_t}$ with n_t samples which are represented by probability distributions p^s and p^t respectively, MMD between p^s and p^t is defined as:

$$\mathrm{MMD}^2(p^s, p^t) = \sup_{||\phi||_{\mathcal{H}} \leq 1} ||\mathbb{E}_{x^s \sim p^s}[\phi(x^s)] - \mathbb{E}_{x^t \sim p^t}[\phi(x^t)]||_{\mathcal{H}}^2 \tag{7.5}$$

where $\mathcal{H}$ is the reproducing kernel Hilbert space (RKHS), $\phi(.)$ is the feature mapping associated with the kernel map $k(x^s, x^t) = < \phi(x^s), \phi(x^t) >$, $\sup(.)$ is the supremum of the input aggregate and $||\phi||_{\mathcal{H}} \leq 1$ defines a set of functions in the unit ball of $\mathcal{H}$. The multi-kernel MMD (MK-MMD) (Gretton et al., 2012) assumes that the optimal kernel is obtained by the linear combination of many kernels. Herein the kernel $k(x^s, x^t)$ is defined as the convex combination of b positive semi-definite kernels $\{k_u\}$ (Gretton et al., 2012):

$$K \triangleq \left\{ k = \sum_{u=1}^{b} \beta_u k_u : \sum_{u=1}^{b} \beta_u = 1, \beta_u \geq 0, \forall u \right\} \tag{7.6}$$

where k is weighted by different kernels and the coefficients β_u is the weight to ensure that the generated multi-kernel k is characteristic. In contrast to MK-MMD which compares all order of statistics, CORAL (Sun et al., 2016) is another discrepancy measure which attempts to align the second-order statistics of the source and target distributions. Deep-CORAL (Sun and Saenko, 2016) extends CORAL for deep neural networks and is defined as follows:

$$\mathrm{CORAL}(x^s, x^t) = \frac{1}{4d^2} ||C_s - C_t||_F^2 \tag{7.7}$$

where $||.||_F^2$ is the squared matrix Frobenius norm, C_s and C_t are the covariance matrices of the source and target domain data. The three losses MK-MMD, CORAL and the supervised classification loss were combined as a *weighted linear combination* to devise the loss function. *Multi-layer domain adaptation* with fully-connected layers fc_1 and fc_2 (shown in

Fig. 7.2b) was performed as it is found empirically in this work to achieve higher target domain accuracy in comparison to single layer adaptation with fc_1 or fc_2. It has been shown in prior works (Yosinski et al., 2014), that adapting a single layer does not sufficiently undo the dataset bias between the source and target domains due to the other non-transferable fc layers. Hence, the proposed $\mathcal{L}_{VTLoss}$ is defined as:

$$\begin{aligned}\mathcal{L}_{VTLoss} = \alpha\mathcal{L}_{crossEnt} + \beta(\{\mathcal{L}^2_{MK-MMD}\}_{fc_1} + \{\mathcal{L}^2_{MK-MMD}\}_{fc_2}) \\ + \lambda(\{\mathcal{L}_{CORAL}\}_{fc_1} + \{\mathcal{L}_{CORAL}\}_{fc_2})\end{aligned} \tag{7.8}$$

where α, β, λ are the hyperparameters. The discrepancy between the source and target domain is reduced by minimising the $\mathcal{L}_{VTLoss}$ as $\min_{G_{vt}(\theta)} \mathcal{L}_{VTLoss}$. The domain adaptation network architecture is shown in Fig. 7.2b.

7.2.4 Deep Active Tactile Object Recognition

Given the trained model using *xAVTNet*, the objective is to classify the object using inference with minimal number of tactile acquisitions actively chosen by the robot as performing tactile actions is time-consuming. A tactile action $\mathbf{a}$ is defined as a ray represented by a tuple $\mathbf{a} = (\mathbf{s}, \overrightarrow{\mathbf{d}})$, with $\mathbf{s}$ as the start point and $\overrightarrow{\mathbf{d}}$ the direction of the ray. The 3D bounding box pose of the object is assumed to be known. The actions are performed as guarded motions so that the robot does not accidentally push or topple the object. The 3D bounding box is discretised into a 3D occupancy grid $\mathcal{OG}$ with resolution g_{res}. Each cell c_i in the occupancy grid is represented by a Bernoulli random variable and has an occupancy probability $p(c_i)$. There are two possible states for each cell with $c_i = 1$ indicating the cell is occupied and $c_i = 0$ for an empty cell. A common independence assumption of each cell with other cells enables the calculation of the overall entropy of the occupancy grid as the summation of the entropy of each cell. The Shannon Entropy of the entire grid can be computed as (Bourgault et al., 2002):

$$\mathbb{H}(\mathcal{OG}) = -\sum_{c_i \in \mathcal{OG}} p(c_i)log(p(c_i)) + (1 - p(c_i))log(1 - p(c_i)) \tag{7.9}$$

To compute the next best touch (NBT), the expected entropy-based information gain is computed. As it is intractable to calculate the exact entropy from a predicted touch, a common simplifying approximation is taken by predicting the expected measurements $\hat{z}_t$ from an action $\mathbf{a}_t$ at time t using ray-traversal algorithms. A virtual sensor model is defined representing the tactile sensor with n_{tax} taxels casting a set of rays $\mathcal{R} = \{r_1, r_2, \ldots r_{n_{tax}}\}$ for a given distance d_{ray} in the *z-axis* of the sensor model coordinate frame, with one ray

Algorithm 5: Overall Algorithm

// Deep Active Visual Recognition
Given: $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$;
Train $G_v(\theta) \leftarrow \mathcal{L}_{crossEnt}$;
if *Classification accuracy* $< \nu$ **then**
  $\mathcal{A}_{view} \leftarrow$ generate_random_views() ;
  ${}^{v}p_{i=1...n_v} \leftarrow$ segment_point_cloud ;
  $D_u \cup {}^{v}p_{i=1...n_v}$;
  $\forall x_u \in D_u$, $p(y|x_u) =$ classify$(G_v(\theta))$;
  $\mathbb{H}(y|\mathbf{x}_u) = -\sum_{c=1}^{C} p(y=c|\mathbf{x}_u) log p(y=c|\mathbf{x}_u)$;
  $D_u \leftarrow$ ranked using $\mathbb{H}$;
  $D_l \leftarrow$ Select top κ% in D_u ;
  $D_l \leftarrow$ annotate_by_human ;
  $D_s \leftarrow D_s \cup D_l$;
// Deep Cross-Modal Visuo-Tactile Transfer Learning
Given: $G_v(\theta), D_s$;
$\mathcal{A} \leftarrow$ generate_random_tactile_actions() ;
$\mathbf{x}^t \leftarrow$ execute_action$(\mathbf{a}_t)$;
$D_t = \{x_j^t\}_{j=1}^{n_t}$;
Train $G_vt(\theta) \leftarrow \mathcal{L}_{VTLoss}$;
// Deep Active Tactile Object Recognition
Given: $\mathcal{T}^{pc} \leftarrow \emptyset$, $G_{vt}(\theta)$;
while $n(\mathcal{T}^{pc}) \leq N_{min}^{T}$ **do**
  $\mathcal{A} \leftarrow$ generate_random_tactile_actions() ;
  $\mathbf{z}_t \leftarrow$ execute_action$(\mathbf{a}_t)$;
  ${}^{t}p \leftarrow \mathbf{z}_t$;
  $\mathcal{T}^{pc} \leftarrow \mathcal{T}^{pc} \cup \{{}^{t}p\}$;
$p(y_t|x_t) =$ classify$(G_{vt}(\theta))$;
while $\arg\max_{c \in C}(p(y_t|x_t) < \zeta$ **do**
  $\mathcal{A}_{nbt} \leftarrow$ generate_possible_actions() ;
  $\hat{\mathcal{Z}} \leftarrow$ pred_measurement$(\mathcal{A})$;
  $\mathbf{a}_{nbt*} \leftarrow$ choose_best_action$(\hat{\mathcal{Z}})$;
  $\mathbf{z}_t \leftarrow$ execute_action$(\mathbf{a}_t^{nbt*})$;
  ${}^{t}p \leftarrow \mathbf{z}_t$;
  $\mathcal{T}^{pc} \leftarrow \mathcal{T}^{pc} \cup \{{}^{t}p\}$;
if $\arg\max_{c \in C}(p(y_t|x_t) > \zeta$ **then**
  $output\_label = \arg\max_{c \in C}(p(y_t|x_t))$;

per taxel. Monte-Carlo sampling of N_{nbt} possible touch points are performed on each face of the bounding box except the bottom face as the object rests on a flat surface. A set of possible touch actions $\mathcal{A}$ is defined for each of the N_{nbt} possible touch points such that the

start points $\mathbf{s}$ are at a fixed distance from the face of the bounding box in the $-z$ axis of the sensor model coordinate frame. The grid cells which are traversed by the rays are computed to be occupied or free and the respective log-odds is updated accordingly (Hornung et al., 2013):

$$l(\hat{z}^{view}) = \begin{cases} log\frac{p_h}{1-p_h} & \text{if } \hat{z}^{view} \widehat{=} \, hit \\ log\frac{p_m}{1-p_m} & \text{if } \hat{z}^{view} \widehat{=} \, miss \end{cases} \tag{7.10}$$

where p_h and p_m are the probabilities of hit and miss which are user-defined values set to 0.7 and 0.4 respectively as in (Hornung et al., 2013). Given the expected observations from all the possible touch points and the updated probabilities of each grid cell, the expected entropy of the overall grid can be evaluated by Eq. (7.9). The expected information gain by taking a touch action $\mathbf{a}$ and corresponding expected measurement $\hat{z}$ is given by the KL divergence between the posterior entropy after integrating the expected measurements and the prior entropy (Potthast and Sukhatme, 2014):

$$E[\mathbb{I}(p(c_i|\mathbf{a}_t, \hat{z}_t))] = \mathbb{H}(p(c_i)) - \mathbb{H}(p(c_i|\mathbf{a}_t, \hat{z}_t)) \tag{7.11}$$

Hence, the selected action $\mathbf{a}^*$ is given by:

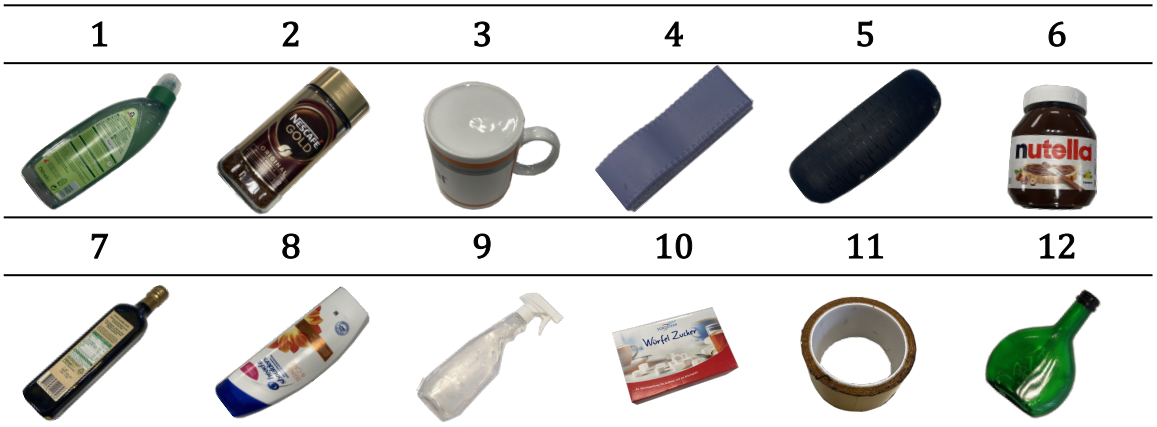
$$\mathbf{a}^* = \arg\max_{\mathbf{a} \in \mathcal{A}} (E[\mathbb{I}(p(c_i|\mathbf{a}_t, \hat{z}_t))]) \tag{7.12}$$

As shown in the Fig. 7.2c, the object classification procedure is started with an initial set of tactile points $\mathbf{p}^{init}$ that are acquired by performing random tactile touch actions. The random tactile touch actions are sampled randomly on the bounding box of the object and performed similarly as actions $\mathbf{a}_t$ explained above. Given minimum number of points for inference (N^T_{min} set to 10 points), the tactile points are collated into a tactile point cloud $\mathcal{T}^{pc}$ and used to perform model inference. If $n(\mathcal{T}^{pc}) < N^T_{min}$ and/or if the output confidence from the model inference is less than a threshold ζ, then active touch actions are performed to acquire additional touch points.

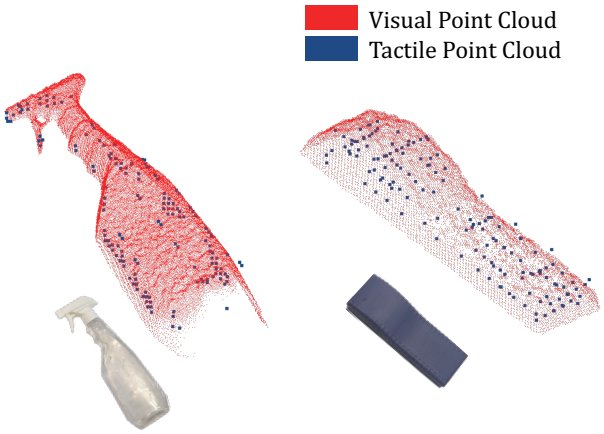
7.3 Experimental Results

7.3.1 Experimental Setup and Data Collection

The experimental setup shown in Fig. 7.1 consists of a Universal Robots UR5 robot with a Robotiq 2F140 Gripper and a Franka Emika Panda robot with the standard Panda Gripper. The Robotiq 2F140 fingertips were equipped with tactile sensor arrays from Xela Robotics as detailed in Chap. 3. The raw data from the Xela sensor consists of a relative value of



(a)

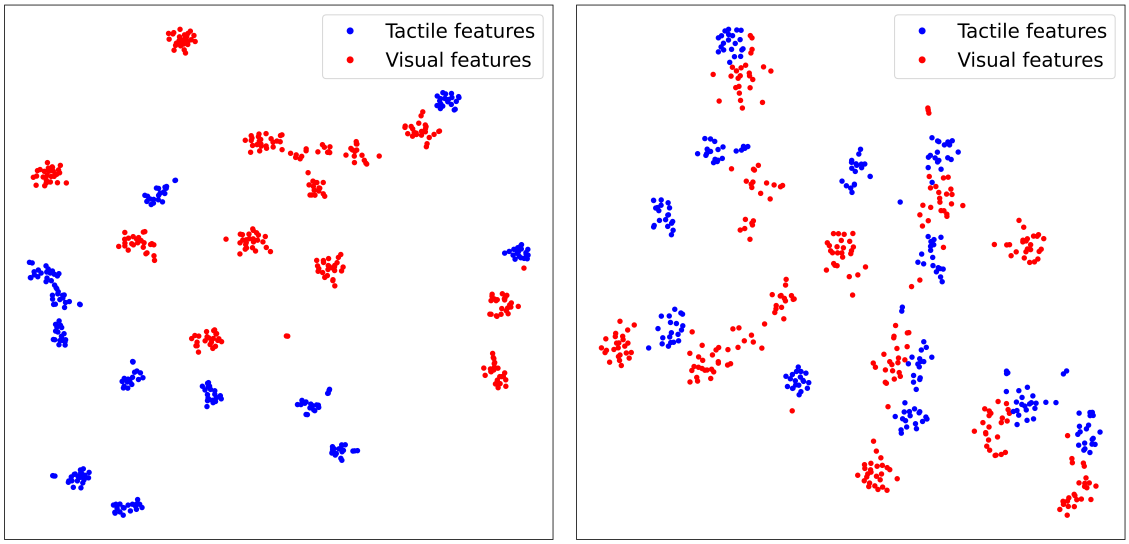


(b)

Figure 7.3: (a) Experimental objects: Twelve daily objects with different characteristic properties such as shape and transparency selected for object recognition task (b) Vision and tactile point clouds of selected objects shown overlapped to demonstrate the difference in point densities.

Table 7.1: The number of labelled samples required to reach a certain relative accuracy measured by the relative error to the fully train network

	No. of Labelled Samples			
Relative error	10%	5%	3%	2%
Random strategy (baseline)	2000	4500	5000	5500
Active strategy (ours)	2000	2000	3000	3500



(a) Before domain adaptation *(b) After domain adaptation*

Figure 7.4: (a) Visual and tactile t-sne features before domain adaptation (b) after performing domain adaptation.

Table 7.2: Ablation study with the domain adaptation methods

Domain Adaptation	**V → T Accuracy (%)**
MMD	57.28 ± 0.77
CORAL	58.34 ± 1.11
VTLoss_{fc1}	70.42 ± 1.23
VTLoss_{fc2}	62.19 ± 2.48
VTLoss	**81.25** ± 1.97

force measurement but its not directly characterised to Newtons. The normal force values (along z-axis) ranges between 36000 and 45000 and the raw force values were normalised. Straight line trajectories with guarded motions were performed to collect the data. When the force value measured on any of the taxels exceeded the threshold $f_r > \tau_f$ (set to 1.1), the motion was stopped and the 3D locations of the contacted taxels were recorded as the tactile point cloud ${}^{t}S$. The tactile point cloud was expressed in the common world coordinate frame $\mathcal{W}$ using the robot's kinematic model. An Azure Kinect DK RGB-D camera was rigidly attached to the Panda Gripper with a custom designed flange which provided the vision point cloud ${}^{v}\mathcal{S}$ in the world-frame using hand-eye calibration. Both visual and tactile point clouds were only composed of the x, y, z coordinates and other properties such as normals, colour are not used. The camera can also be used to extract the bounding box

Table 7.3: Confusion matrix for tactile object recognition

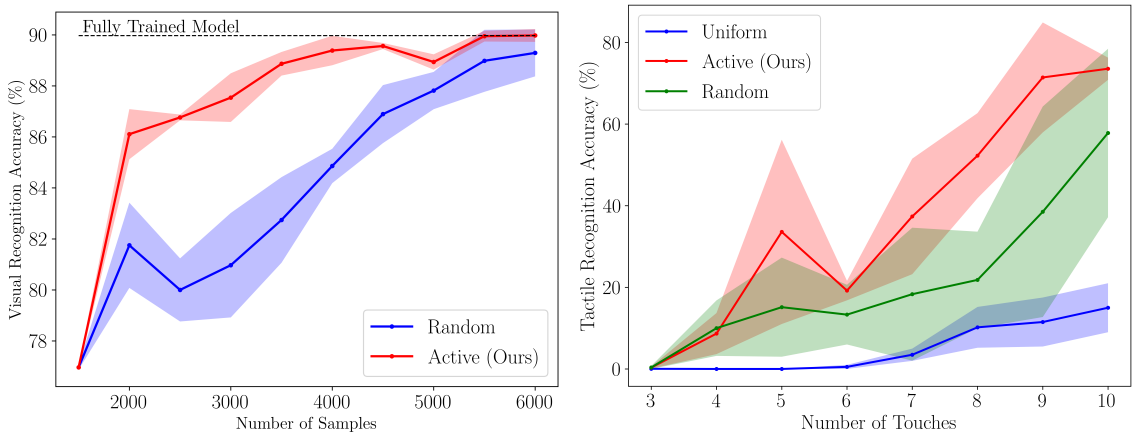
1	0.85	0	0	0	0	0	0	0.05	0.1	0	0	0
2	0.2	0.75	0	0	0	0	0.05	0	0	0	0	0
3	0	0	0.95	0	0	0.05	0	0	0	0	0	0
4	0	0	0	1	0	0	0	0	0	0	0	0
5	0.1	0	0	0.05	0.8	0	0.05	0	0	0	0	0
6	0	0	0	0	0	1	0	0	0	0	0	0
7	0	0	0	0	0	0	0.9	0	0	0	0	0.1
8	0.7	0	0	0	0	0	0	0.2	0	0	0.1	0
9	0.05	0	0	0	0	0	0	0	0.95	0	0	0
10	0.05	0	0	0	0	0	0	0.1	0	0.45	0.4	0
11	0	0	0	0	0	0	0	0	0	0	1	0
12	0	0	0	0	0.05	0	0	0	0.05	0	0	0.9
Object	1	2	3	4	5	6	7	8	9	10	11	12

Table 7.4: Comparison study with state-of-the-art approach

Method	**V → T Accuracy (%)**
CLUE+GFK+1NN (Falco et al., 2019)	26.72 ± 0.96
CLUE+GFK+SVM (Falco et al., 2019)	15.05 ± 2.54
xAVTnet-noDA	51.25 ± 2.25
xAVTNet (ours)	**81.25** ± 1.97

pose of the object using point cloud segmentation and clustering methods from the Point Cloud Library (Rusu and Cousins, 2011). The OctoMap library (Hornung et al., 2013) was used for the next best touch implementations. A ROS-based framework was used for controlling the robots, sensor acquisitions and data collection.

Network Implementation: The PointNet (Qi et al., 2017) network was used which performs input and feature transformations to encode a 1024 global feature vector. It's followed by three fully connected layers fc_1,fc_2,fc_3 of size 512, 256 and k respectively. The hidden layers fc_1, fc_2 include ReLU and batch normalisation. Furthermore, the dropout with probability 0.4 was used on the fc_2 layer. ADAM optimiser was used and learning rate was set to 10^{-3}. Two streams of fc layers were used for domain adaptation as shown in Fig. 7.2b. The hyper-parameters were empirically tuned: $\alpha = 10$, $\beta = 10$ and $\lambda = 10$. The robot experiments were performed on a workstation running Ubuntu 18.04 with 8 core Intel i7-8550U CPU @ 1.80GHz and 16 GB RAM. The training and domain adaptation of the network was performed using PyTorch framework on a workstation with NVidia Quadro RTX 4000 GPU with 8 GB RAM.



(a) Vision: Active learning *(b) Tactile: Active recognition*

Figure 7.5: (a) Active strategy versus random strategy for deep visual learning (solid line: mean, shaded: standard deviation). (b) Active strategy versus uniform and random strategy for tactile object recognition (solid line: median, shaded: median absolute deviation).

A set of 12 real world objects were used for the task of object recognition as shown in Fig. 7.3a. The objects were ordered in a list as follows: detergent bottle, coffee bottle, mug, whiteboard cleaner, box, nutella bottle, olive oil bottle, shampoo bottle, spray bottle, sugar box, tape and green bottle. The objects were selected based on varying degree of shape complexity and transparency that is challenging for visual sensors. The visual and tactile point clouds of some objects are shown in Fig. 7.3b highlighting the difference in the number of points and point density between the two domains. Although dynamic viewpoints were used for the visual point cloud acquisition, some regions of the objects remain occluded due to the kinematic limits of the fixed-base manipulator. A point cloud from a viewpoint was considered as one training sample and multiple viewpoints were not merged together. The visual sensor may also produce noisy measurements and warped point clouds due to acute viewing angles which have been retained to make the network robust to real-world sensors.

7.3.2 Robot Experiments

Deep Active Visual Object Learning: Initially if the scene was cluttered, the autonomous decluttering technique presented in Chap. 5 can be used. After the scene has been decluttered, the Panda robot initiated visual data collection. A total of 9300 visual point clouds for the 12 objects were collected by autonomously commanding the Panda robot to different viewpoints. The dataset included the data augmentation performed by random rotations around the Z-axis to be rotation invariant. The scene was also manually rearranged between data collection iterations to capture all possible views of the object.

However, it should be noted that the model was not affected by the relative pose of the objects. For training, 6000 point cloud samples were used, 1500 for validation and 1800 for the test set. Initially, all training samples are unlabelled and represent the unlabelled dataset D_u. Subsequently, 1500 samples from D_u were randomly selected and the human user (oracle) labelled them to train the network. The trained network was used to compute the uncertainty of the remaining unlabelled samples as explained in Sec. 7.2.2. The proposed active learning strategy has been compared with a baseline method that randomly queries samples from D_u in this section. At each query step, $\kappa = 500$ samples were queried from the unlabelled dataset. The mean (solid line) and standard deviation (shaded region) results for deep active visual object learning versus baseline are presented in Fig. 7.5a. The number of labelled samples necessary to achieve a certain relative error of the fully trained network has been reported in Tab. 7.1.

Visuo-Tactile Domain Adaptation (DA): For DA, 10 tactile point clouds for each object were collected using random tactile collection strategy. Similarly to the visual dataset, the tactile dataset was augmented by performing random rotations around the Z-axis to be rotation-invariant, increasing the data set to 100 point clouds per object. All tactile point clouds were unlabelled because the objective was to perform unsupervised domain adaptation. For domain adaptation training, 900 samples were used and 300 samples were used as a test set. Tab. 7.3 shows the confusion matrix for classification accuracy in the test set after performing unsupervised domain adaptation using the proposed *VTLoss* function. In order to show the performance of domain adaptation method, comparison has been performed against the MMD loss and the CORAL loss as ablation studies shown in Tab. 7.2. Since multilayer domain adaptation has been used, the other variants in which a single fully connected layer $fc1$ or $fc2$ was used for domain adaptation are reported in Tab. 7.2.

To benchmark the proposed framework, comparison against a state-of-the-art visuo-tactile cross-modal domain adaptation work of Falco et al. (2019) has been performed. Due to unavailability of official source code from their work, their paper has been reimplemted as follows: A hand-crafted feature descriptor termed CLUE (Cross modal point cLoUd dEscriptor) was proposed which has been implemented using PCL and the geodesic flow kernel (GFK) for domain adaptation using MATLAB domain adaptation toolbox (Yan, 2024). The k-nearest neighbours (kNN) classifier and support vector machines (SVM) were used for classification and the parameters have been fine-tuned according to guidelines in (Falco et al., 2019). It must be noted that the re-implementation shown here may not be identical to that of the original implementation. The same objects from Fig. 7.3a were used for the evaluation. Importantly, the dense visual and sparse tactile point clouds were used directly without equalising the point cloud density of the two domains as done in (Falco

et al., 2019). The test classification accuracy with tactile data is reported in Tab. 7.4.

Deep Active Tactile Object Recognition: The method proposed in this chapter is independent of the method of tactile data collection. The data can be recorded uniformly, randomly or even actively exploiting information gain. In order to evaluate the proposed active tactile recognition method, comparison has been performed against uniform and random collection strategies. The uniform collection strategy is defined as follows: the 3D bounding box around the object is discretised into a grid of cell size 3 cm $\times$ 3 cm corresponding to the size of the tactile sensor array. The cells are explored sequentially starting from the edge closest to the robot base. The random strategy follows similarly to the active strategy, with the next touch chosen randomly among possible touch actions instead of using information gain metric. In order to fairly compare the strategies, only the first 10 touches were selected. The parameter N^T_{min} is set to 10 points in the experiments beyond which active, uniform, or random acquisition was performed. The acquisition strategies were compared from the third touch and the median (solid line) and median absolute deviation (shaded region) are shown in Fig. 7.5b. The confidence threshold ζ to stop active tactile exploration for recognition was set at 0.8 or 80%. On average, the active approach took around 14 tactile actions, the random approach around 19 actions, and the uniform approach took 27 actions to achieve a classification precision over the confidence threshold.

7.4 Discussion

As seen from Tab. 7.3, the proposed network has an average accuracy of 81.25%. The network has an accuracy over 80% for 9 out of 12 objects. The objects with lower accuracy include (i) object 2 (coffee bottle) at 75%, (ii) object 8 (shampoo bottle) at 20% and (iii) object 10 (sugar box) at 45%. The shampoo bottle was confused with object 1 (detergent bottle) due to the similar shape and curvature. In fact, if the tactile sensor does not acquire data around the head of the two bottles, due to the sparsity of the tactile data, the trained model was confused. The sugar box was confused with the tape (object 11). Although the shapes were different, the inaccuracies were due to the fact that the rigidity of the tactile sensor array does not accurately capture high curvatures present in the tape. The sugar box also undergoes minor deformations while performing tactile data acquisitions. From Tab. 7.4 it can be noted that the proposed approach outperforms the state-of-the-art method (Falco et al., 2019) by over 50%. The reduced accuracy of (Falco et al., 2019) was due to the fact that an important assumption in their work was relaxed by using dense visual point clouds and sparse tactile point clouds directly without equalising the number of points. In addition, the baseline method (Falco et al., 2019) was developed primarily for quasi-planar objects while this work used a dataset comprised of 3D objects of varying

shape complexity. Furthermore, the deep neural network was able to extract the discernible features from even sparse point sets by transferring knowledge gained from dense point clouds that hand-crafted features extractors such as CLUE (Falco et al., 2019) fail to do so. Using the proposed cross-modal transfer learning technique, an improvement of accuracy of nearly 30% over the same network without domain adaptation is seen demonstrating the efficacy of the method. Furthermore, the domain adaptation method *VTLoss* combining MMD, CORAL and the classification loss in a weighted linear combination outperforms both MMD and CORAL by over 20%. Similarly, the multi-layer adaptation provides an improved performance of over 10% compared to single-layer adaptation. From the t-distributed stochastic neighbor embedding (t-SNE) (Van Der Maaten, 2014) visualisations from Fig. 7.4 it can be seen that the source (visual) features and the target (tactile) features are well clustered after applying domain adaptation. This shows that the model has learnt to effectively discriminate the target features without explicitly training with labelled target data. The proposed framework is also data efficient. The Tab. 7.1 shows that the active learning approach demonstrates high accuracies within 5% relative error or 2% relative error to that of a fully trained model using just 33% and 58% of the complete dataset respectively. Fig. 7.5a shows that the proposed active learning strategy outperformed the baseline random query strategy for visual object learning with fewer data. This demonstrates the amount of labelling efforts saved by adopting the active learning strategy. Similarly, the active tactile object recognition method outperformed the uniform action strategy as seen from Fig. 7.5b. Using the active tactile approach, the robot can recognise objects with $> 70\%$ accuracy whereas a uniform strategy only reached 20% accuracy within the first 10 touch actions. The random strategy reached around 60% accuracy while having larger variability of the recognition as expected from a randomised approach. This helped reduce the overall time for the task execution as robotic tactile action execution is time-consuming.

To summarise, this chapter tackled the problem of robotic visuo-tactile cross-modal object recognition leveraging deep neural networks and active perception and learning. The presented network *xAVTNet*, actively learnt from labelled visual point cloud samples and unsupervised cross-modal transfer learning was performed with unlabelled tactile point clouds using the novel domain adaptation *VTLoss* function. The cross-modal transfer learning method outperformed state-of-the-art approaches in cross-modal object recognition accuracy. The proposed active perception and learning methods also demonstrated clear outperformance over baseline strategies leading to a reduction in human labelling effort and a faster data collection time. Furthermore, the proposed framework used an active tactile object recognition strategy which led to data efficiency by reaching high accuracy with fewer data collection steps.

Considering real-world applications, the cross-modal recognition approach finds potential applications in robotic operations in adverse and unknown environments. In situations where visual sensing is impeded by factors such as smoke, dust, and other factors, the robot is effectively operating in a blind manner. Through cross-modal vision-to-tactile transfer learning, the robot can recognise objects through sense of touch, and thereby increase the resilience of the system. Similarly, tasks that involve object retrieval and manipulation in cluttered settings, such as bin picking, can greatly benefit from these techniques. When a robotic gripper is engaged in grasping objects within densely cluttered environments, a direct line-of-sight to the robot is often obstructed. Consequently, the robot must be entirely dependent on tactile sensing on the gripper to identify and retrieve the object. Furthermore, the learnt visuo-tactile latent space encapsulates comprehensive information essential for encoding tactile signals predicated on visual inputs (for instance, the tactile sensory output associated to an image showing a velvet-like fabric), thereby facilitating its application within virtual reality (VR) settings (Maier et al., 2016).

Part IV

Category-Level Perception for Object Reconstruction

Chapter 8

Active Tactile based Category-Level Object Reconstruction

Parts of this chapter are **published** as:

- "*Touch if it's transparent! ACTOR: Active Tactile-based Category-Level Transparent Object Reconstruction*", **P. K. Murali**, B. Porr, and M. Kaboli, In 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 10792-10799) (Murali et al., 2023).

The video of the experiments from this chapter is available here:
`https://drive.google.com/file/d/1H6wQRfXTGzV3UIvxfjiYo1svmeap8TPk/view?usp=sharing`

8.1 Introduction

Reconstruction of shapes of unknown objects is a fundamental perception task that enables downstream tasks for robots such as pose estimation, grasping and manipulation and so on. Transparent objects such as cups, glasses, and bottles are ubiquitous around us and if robots are expected to work in unstructured scenarios such as household environments, it is essential to recognise and safely manipulate transparent objects. Although the reconstruction task is relatively simpler for opaque objects with off-the-shelf vision sensors, such sensors produce unreliable and erroneous data with transparent objects due to their non-Lambertian surfaces. Sophisticated custom-calibrated sets with specialised scanners or modifying the transparent surface of objects are often necessary for accurate reconstruction (Ihrke et al., 2010). This is not practical for robots that need to perform reconstruction on-the-fly for arbitrary unknown objects. However, high-fidelity tactile sensing can be used

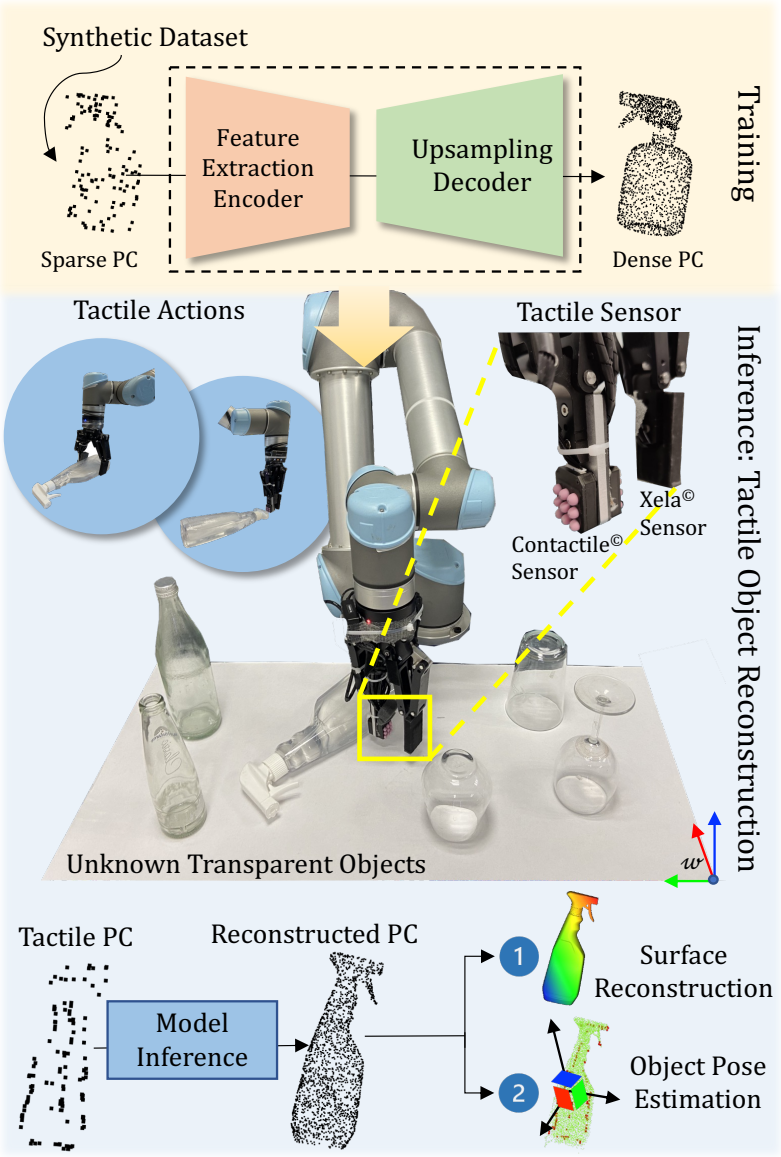


Figure 8.1: Experimental Setup: A Universal Robots UR5 with sensorised Robotiq Gripper with 3-axis tactile sensor arrays for active tactile-based category-level unknown transparent object reconstruction.

for the reconstruction of the shape of transparent objects, as well as perform pose estimation and safe manipulation (Li et al., 2020a; Kaboli et al., 2016).

Tactile perception is inherently action-conditioned as data depends on the type of contact action performed and local as only the local surface information around the contact area is extracted. Hence, for reconstructing the surfaces of an object, multiple contact actions need to be performed by the robot. This leads to sparse information and prohibitively long data collection times. The prior works in literature have worked towards addressing some

of these challenges. Early works have used offline methods to collect dense tactile data and fit to shape primitives such as superquadratics (Bierbaum et al., 2007). Reconstruction of 2-D shapes of convex objects with simple shapes using tactile sensing was performed by deriving a closed-form solution for the curvature at the contact point and rotational speeds in (Moll and Erdmann, 2001). Aggregating contact points into a point cloud is often used to represent the shape of the objects. Some works have used Bayesian filtering techniques for defining a probabilistic model of the objects using the tactile point clouds and used them for other tasks such as classification (Meier et al., 2011). Teleoperation-based sliding of tactile sensor against an object surface and using contour fitting techniques to reconstruct the surface has been used in (Jia and Tian, 2009). Gaussian process implicit surface (GPIS) has been widely used for tactile object reconstruction (Dragiev et al., 2011; Yi et al., 2016; Björkman et al., 2013; Gandler et al., 2020; Martens et al., 2016; Suresh et al., 2021; Jamali et al., 2016). The implicit surface described by a Gaussian process describes the shape of an object through a function that decides for each point in space whether it is part of the object or not. It produces smooth surface manifolds with a reasonable number of tactile points as input and also provides probabilistic information to guide the tactile actions. However, for complex shapes it typically requires lots of points uniformly distributed on the object's surface for reconstruction (Rustler et al., 2022; Jamali et al., 2016). Some works have also used tactile sensing with visual perception to complete the shape with prior information observed with visual camera (Gandler et al., 2020; Rustler et al., 2022; Smith et al., 2020). While these works focus on opaque objects, limited works exist for the reconstruction of transparent objects. Recently, deep learning methods have been used for point cloud based shape completion given partial or noisy input point clouds (Fei et al., 2022). Seminal works on PointNet (Qi et al., 2017) allowed using raw point clouds as inputs to deep networks for the task of classification and semantic segmentation. Prior works have worked towards point cloud completion using deep networks such as (Yu et al., 2018; Yuan et al., 2018) but are mainly evaluated on datasets derived from CAD models and rarely evaluated on real-world platforms with noisy and sparse sensors (Fei et al., 2022).

There are limitations in the state-of-the-art for the reconstruction of photometrically challenging objects such as transparent and shiny objects with tactile perception: (a) existing reconstruction strategies such as Gaussian Process Implicit Surfaces (GPIS) fail to capture fine shape details with sparse tactile input data, and (b) directly deploying deep learning based strategies for shape completion with sparse input data is impractical as the collection of a large dataset of tactile data for training is prohibitively expensive.

The contributions of this chapter are as follows:

(I) A novel framework for deep active tactile-based category-level reconstruction of un-

known objects. The proposed reconstruction network has been trained on a category-level synthetic dataset and tested on sparse tactile point clouds from real unknown transparent objects.

(II) An autonomous and active tactile-based unknown object exploration strategy based on information gain from sampling possible tactile actions (such as probing and pinch grasp) leading to improved data collection efficiency.

To validate the proposed framework, extensive experiments have been performed on a real robotic setup shown in Fig. 8.1 and compared against state-of-the-art methods.

8.2 Methodology

8.2.1 Problem Definition and Proposed Framework

The objective was to reconstruct a dense point cloud that precisely represented the shape of the unknown transparent objects from sparse point clouds extracted with active tactile interactive perception (touch/ pinch). To this end, a novel framework termed ACTOR was proposed as shown in Fig. 8.2. In Fig. 8.2(a) a self-supervised learning approach with an auto-encoder network is presented that was trained on subsampled point clouds from synthetic objects belonging to the same category but not identical as the real objects. In Fig. 8.2(b), a novel active tactile-based unknown transparent object exploration strategy is shown for inference with the trained model to reconstruct a dense point cloud. Further downstream tasks such as tactile-based object recognition was also demonstrated from the reconstructed model.

8.2.2 Deep Self-Supervised Learning for 3D Object Reconstruction

A dataset $\mathcal{D}$ was constructed based on synthetic object models from the ShapeNet repository (Chang et al., 2015) in order to leverage the synthetic open-source datasets and avoid expensive real tactile-data collection (Murali, 2025). The synthetic object models belonged to the same category but were different from the real unknown transparent objects. Point clouds of size $N_{in} = 2048$ points were sampled from the synthetic object meshes. These point clouds were normalised and scaled to fit into a $[0,1]^3$ cube and added to the dataset, $\mathcal{P}_{in} \in \mathcal{D}$. In order to generate the input point clouds $\mathcal{P}^{\bullet}_{in}$ to the network, the point clouds $\mathcal{P}_{in}$ were randomly subsampled by voxel-grid subsampling by the factor k i.e., $\mathcal{P}^{\bullet}_{in} \in \mathbb{R}^{\lceil \frac{1}{k} N_{in} \rceil \times 3}$. This creates a challenging task for reconstruction with higher values for k as simpler techniques based on interpolation with neighbourhood points cannot be used.

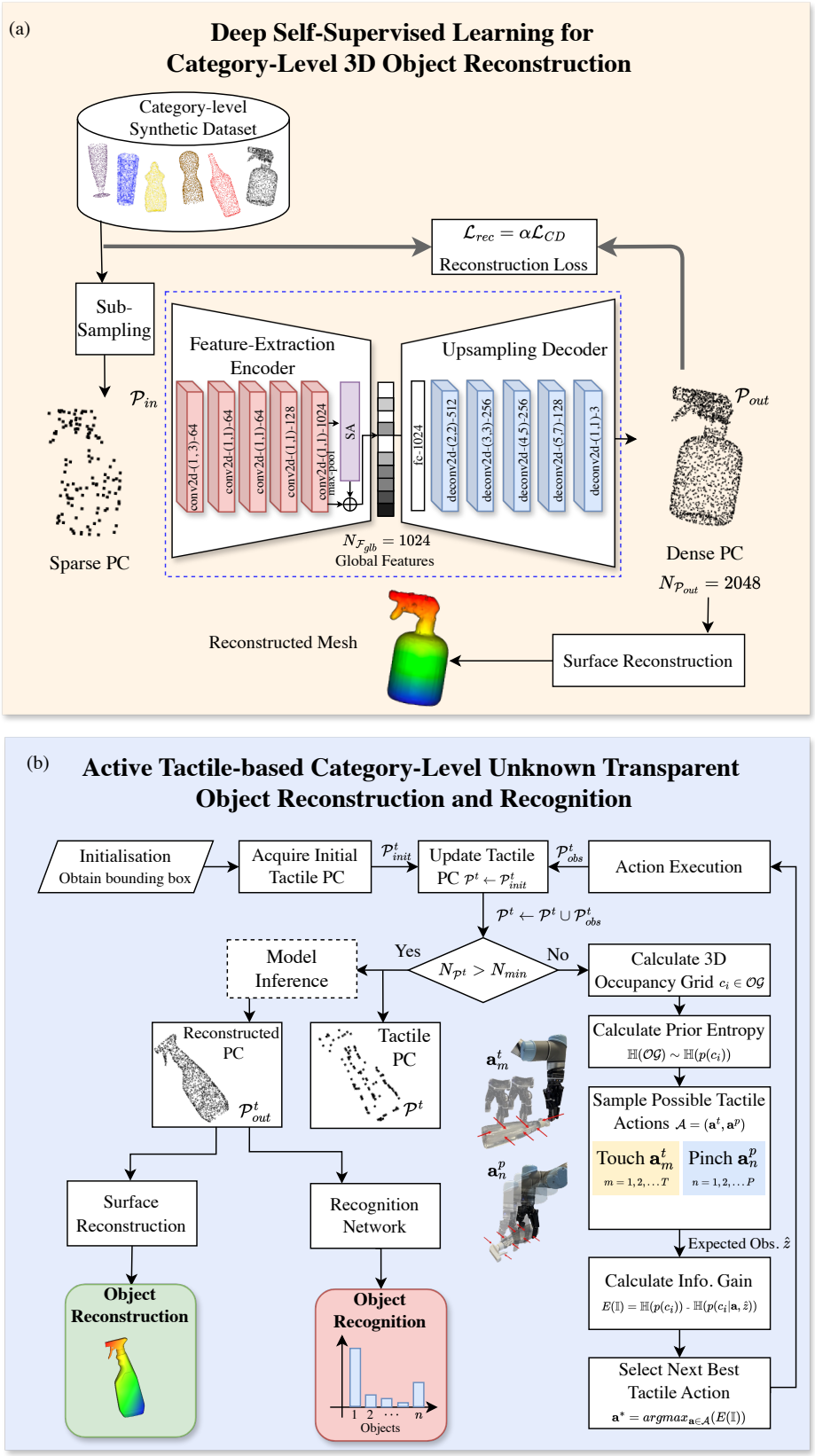


Figure 8.2: Proposed framework Active Tactile-based Category-Level Transparent Object Reconstruction.

Feature-Extraction Encoder The network architecture shown in Fig. 8.2(a) is an auto-encoder that uses a self-supervised learning approach to reconstruct the original point cloud from a subsampled input point cloud. The encoder takes subsampled point clouds as inputs and generates a high dimensional feature vector. The feature vector captures the global geometric shape information of the input point cloud. In general, any deep network that works on raw input point clouds to provide a high dimensional feature vector can be used as an encoder. In this case, a modified PointNet architecture (Qi et al., 2017) was used for the encoder. The encoder network takes unordered point clouds and generates a global feature descriptor vector of size 1024. The network learns a set of optimisation functions that selects informative point regions in the point cloud. The encoder consists of $[1 \times 1]$ convolutions with output channels size $(64, 64, 128, 1024)$ with the first convolutional layer with kernel size $[1 \times 3]$ to encode the input point cloud of $N \times 3$ dimension. The convolution layers were aggregated by a max-pooling layer. A self-attention layer (Zhang et al., 2019) was introduced whose outputs were aggregated with the max-pooled features to provide the global feature vector. The encoder architecture is summarised in Fig. 8.2a. As the encoder provides a high-dimensional global feature vector, it's termed as feature-extraction encoder.

Self-Attention Layer: The self-attention layer was introduced as it can encode meaningful spatial relationships between features and focus on important local features. Two separate multi-layer perceptrons (MLPs) were used to obtain features $\mathbf{G}$ and $\mathbf{H}$ which were subsequently used to get the weights as $\mathbf{W} = softmax(\mathbf{G}^T\mathbf{H})$. The input features were transformed using another MLP to obtain $\mathbf{K}$ and multiplied with the weights as $\mathbf{W}^T\mathbf{K}$. These vectors were summed with the input vector to produce the output features. The self attention layer description is shown in Fig. 8.3.

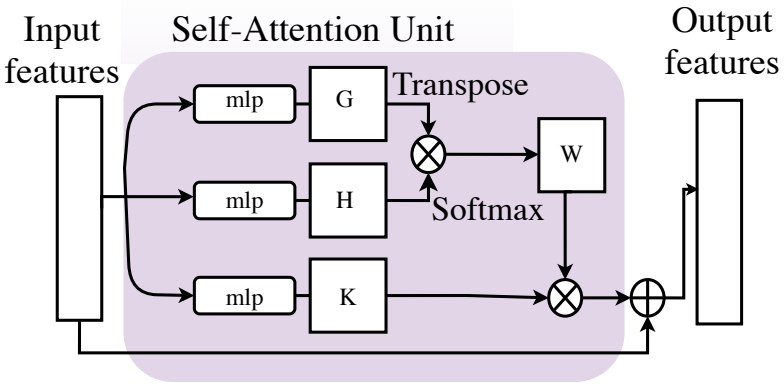


Figure 8.3: The self-attention unit.

Upsampling Decoder An upsampling decoder was developed that upsampled the input global feature vector to provide the reconstructed dense output point cloud $\mathcal{P}_{out}$. The upsampling decoder was composed of a fully connected layer with output dimension of 1024 and five deconvolutional layers or transpose convolutional layers with kernel sizes and output channels shown in Fig. 8.2a. The decoder produced the output point cloud with point size set to 2048 while training as this was sufficiently dense for reconstruction purposes.

Loss Function In order to encourage the upsampled point cloud to be close to the original input point cloud and follow the underlying geometrical surface of the object, the Chamfer distance metric (Borgefors, 1986) was used as the loss function. Given the input point cloud to the proposed network prior to subsampling, $\mathcal{P}_{in}$ and the reconstructed output point cloud $\mathcal{P}_{out}$, the loss is defined as:

$$\mathcal{L}_{CD}(\mathcal{P}_{in}, \mathcal{P}_{out}) = \frac{1}{|\mathcal{P}_{in}|} \sum_{p_1 \in \mathcal{P}_{in}} \min_{p_2 \in \mathcal{P}_{out}} ||p_1 - p_2||_2 + \frac{1}{|\mathcal{P}_{out}|} \sum_{p_2 \in \mathcal{P}_{out}} \min_{p_1 \in \mathcal{P}_{in}} ||p_2 - p_1||_2 \tag{8.1}$$

where $|\bullet|$ refers to the number of points in the point cloud and $||\bullet||_2$ refers to the L2 norm. The loss $\mathcal{L}_{CD}$ represents the average distance between the *closest* points in the point clouds. A weighted loss was used for learning stability for the reconstruction loss as $\mathcal{L}_{rec} = \alpha \mathcal{L}_{CD}$ with $\alpha = 100$ set empirically.

Recognition Network The pretrained encoder layers for reconstruction task were frozen for the task of category-level classification. Three fully-connected layers with parameters 512, 256, and n respectively where n represents the number of categories of the objects were used for the classification task. The softmax cross-entropy loss was used for training the recognition network. The subsampled sparse point clouds from the synthetic dataset with different subsampling ratios and data augmentation with random rotations were used. Network implementation details are provided in Sec. 8.3.1. For surface reconstruction from the dense reconstructed point cloud, the ball-pivoting algorithm (Bernardini et al., 1999) was used.

8.2.3 Active Deep Tactile-based Unknown Transparent Object Reconstruction

The model which was trained with only synthetic data was used during the inference with real-world transparent objects. The sparse tactile point cloud data was collected au-

tonomously by the robot using an information gain-based active strategy. Two types of tactile actions were used for data acquisition: touch and pinch actions as shown in Fig. 8.4. The touch actions were executed as guarded horizontal straight-line motions wherein the object was not moved upon contact. The touch action is defined by a tuple $\mathbf{a}^t = \{\mathbf{s}^t, \overrightarrow{\mathbf{d}^t}\}$ where $\mathbf{s}^t \in \mathbb{R}^3$ is the start point of the tactile-sensorised gripper and $\overrightarrow{\mathbf{d}^t} \in \mathbb{R}^3$ is the direction of the gripper-motion defined in the world-coordinate frame $\mathcal{W}$. During the pinch action, the robot approached the object in a vertical straight-line motion with a completely open gripper and performed an antipodal enclosure grasp on the object without moving the object. The fingers of the gripper were closed until the force on the tactile sensors exceeds a predefined threshold $f^p > \tau$. The pinch action is defined as $\mathbf{a}^p = \{\mathbf{s}^p\}$ where $\mathbf{s}^p \in \mathbb{R}^3$ is the start position of the gripper motion vertically above the object at a predefined height as shown in Fig. 8.4. Given the 2D bounding box of the object (*a priori* known or through a RGB-image sensor), a probabilistic occupancy grid $\mathcal{OG}_i$ of preset height and resolution og_{res} was defined. Each cell of the occupancy grid c_i was represented by an occupancy probability $p(c_i)$ which was initially set to 0.5. During exploration, if a cell is discovered to belong to the object, the probability is set to 1 and similarly, if the cell belongs to free space, the probability is set to 0. The probabilities are updated through ray intersections based on the virtual sensor model. A virtual sensor model of the tactile sensor was defined which casted a set of rays $\mathcal{R} = \{r_1, r_2, \ldots, r_{n_{taxel}}\}$ where n_{taxel} refers to the number of taxels in the sensor array. The independence assumption of the probability of each grid cell with one another allows to calculate the overall entropy of the $\mathcal{OG}$ as the summation of the entropy of each cell. The Shannon entropy of the overall occupancy grid can be calculated as:

$$\mathbb{H}(\mathcal{OG}) = \sum_{c_i \in \mathcal{OG}} p(c_i) log(p(c_i)) + (1 - p(c_i))(1 - log(p(c_i))) \tag{8.2}$$

For performing the next best tactile (NBT) action, Monte-Carlo sampling of possible tactile actions N_{nbt} were performed. The action space $\mathcal{A}_{nbt}$ was comprised of an equal number of touch and pinch respectively as $\mathcal{A}_{nbt} = \{a^p, a^t\}_{N_{nbt}}$. The expected measurements $\hat{\mathbf{z}}_t$ for each action $a_t \in \mathcal{A}$ were computed using ray-traversal algorithms (Hornung et al., 2013). Given the observed grid cell c and the measurement from sensor observation z, the log-odds were updated as $L(c|z) = L(c) + l(z)$ wherein $L(c) = log\frac{p(c)}{1-p(c)}$ and

$$l(z) = \begin{cases} log\frac{p_h}{1-p_h} & \text{if } z \widehat{=} \, hit \\ log\frac{p_m}{1-p_m} & \text{if } z \widehat{=} \, miss \end{cases} \tag{8.3}$$

where p_h and p_m are the probabilities of hit and miss which are user-defined values set to 0.7 and 0.4 respectively as in (Hornung et al., 2013). The posterior probability $p(g|z)$

can be computed by inverting $L(c|z)$. The expected information gain by taking an action $a_t \in \mathcal{A}_{nbt}$ with expected measurement $\hat{\mathbf{z}}_t$ is provided by the KL divergence of the posterior entropy and the prior entropy as:

$$E[\mathbb{I}(p(c_i|\mathbf{a}_t,\hat{z}_t))] = \mathbb{H}(p(c_i)) - \mathbb{H}(p(c_i|\mathbf{a}_t,\hat{z}_t)) \tag{8.4}$$

Therefore, the NBT action was the action that maximised the expected information gain as:

$$\mathbf{a}_t^{nbt*} = \arg\max_{\mathbf{a}\in\mathcal{A}}(E[\mathbb{I}(p(c_i|\mathbf{a}_t,\hat{z}_t))]) \tag{8.5}$$

Each tactile action extracted contact positions in 3D space and contact forces. The direction of the normal force was used to extract the normal direction $\hat{n}$ of the object surface. The contact points were aggregated into the tactile point cloud $\mathcal{P}^t$. In order to initialise the NBT action calculation, an initial point cloud (with $N_{\mathcal{P}^t} = 20$) was extracted by randomised touch actions. Further points were collected in an active manner using the NBT criteria. A minimum number of points in the tactile point cloud were required to perform model inference $N_{\mathcal{P}^t} > N_{min}$ which was set empirically. The tactile point cloud was provided as input to the trained network and the reconstructed point cloud $\mathcal{P}_{out}$ was obtained as the output. It may be noted that the reconstructed point cloud may be used for performing pose estimation with the S-TIQF algorithm presented in Chap. 5. The global feature vector in the latent space was used for object recognition with the recognition network defined in Sec. 8.2.2.

8.3 Experimental Results

8.3.1 Experimental Setup

The experimental setup is shown in Fig. 8.1 consists of a set of 9 unknown transparent objects belonging to six categories and a Universal Robots UR5 equipped with a sensorised Robotiq 2F140 gripper. The Contactile and Xela tactile sensors were used as explained in Chap. 3. The normalised force values of the tactile sensors were measured and contact was established when the force exceeds the baseline threshold $f_{ts} \geq \tau_f$ where $\tau_f = 1.1$. The contact points P^t_{obs} were added to the tactile point cloud P^t after every action. The tactile point cloud P^t consisted of x,y,z positions and the normal direction $\hat{n}$ extracted from the normal force vector. The normal information was only used for the baseline GPIS computation and the surface reconstruction. All operations involving point clouds used the Point Cloud Library (Rusu and Cousins, 2011), occupancy grid computations uses Octomap library (Hornung et al., 2013), and the overall setup used the ROS Melodic-based

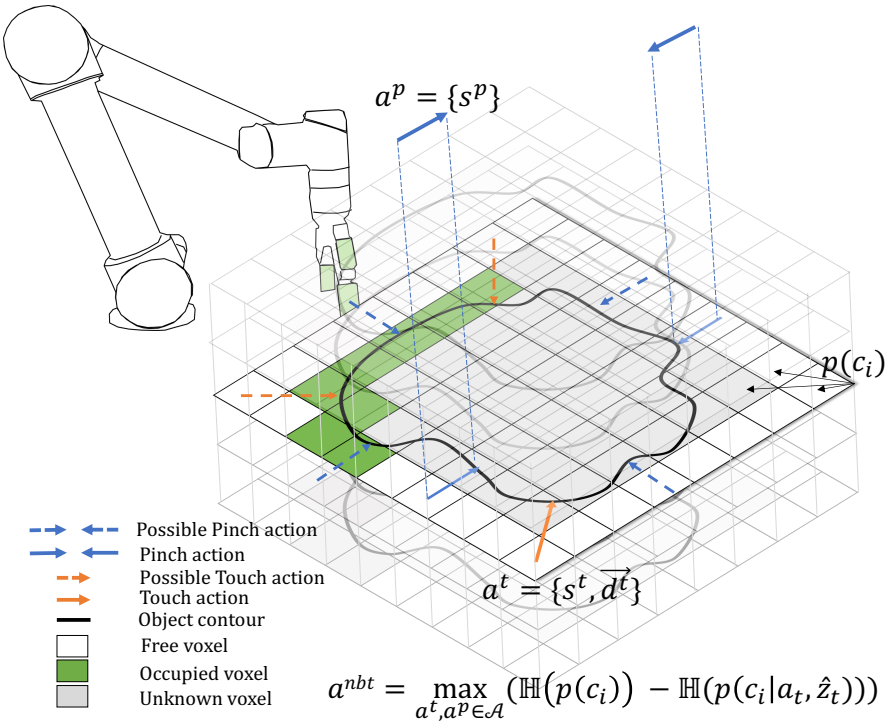


Figure 8.4: Action selection voxelised probabilistic occupancy grid.

middleware.

Network Implementation Details

The proposed reconstruction network described in Sec. 8.2.2 was implemented using Tensorflow framework and training/ inference were performed on Nvidia Quadro RTX 4000 GPU. The network training used ADAM optimiser, learning rate set to 10^{-4}, momentum 0.9 and batch size 8. All layers of the encoder-decoder used batch normalisation and the decay rate initialised at 0.5 and gradually increased to 0.99 with decay step size 2×10^5. During training with the synthetic dataset $\mathcal{D}$, random voxel-grid subsampling was done to have input point clouds with point size between 40 and 120. The hyper-parameter α for the loss was set to 100. For the recognition head, a dropout with keep probability 0.7 was used on the $\texttt{fc} - 256$ layer.

Object List

The following widely-available transparent objects were used for reconstruction: bottle 1, bottle 2, can, detergent, cup 1, cup 2, cup 3, wineglass and spray as shown in Fig. 8.6. The object nomenclature follows $\{$*category-name*$\}$ $\{$*instance number*$\}$. The real objects were chosen to have varying complexities in shape from simple (bottle) to complex (spray).

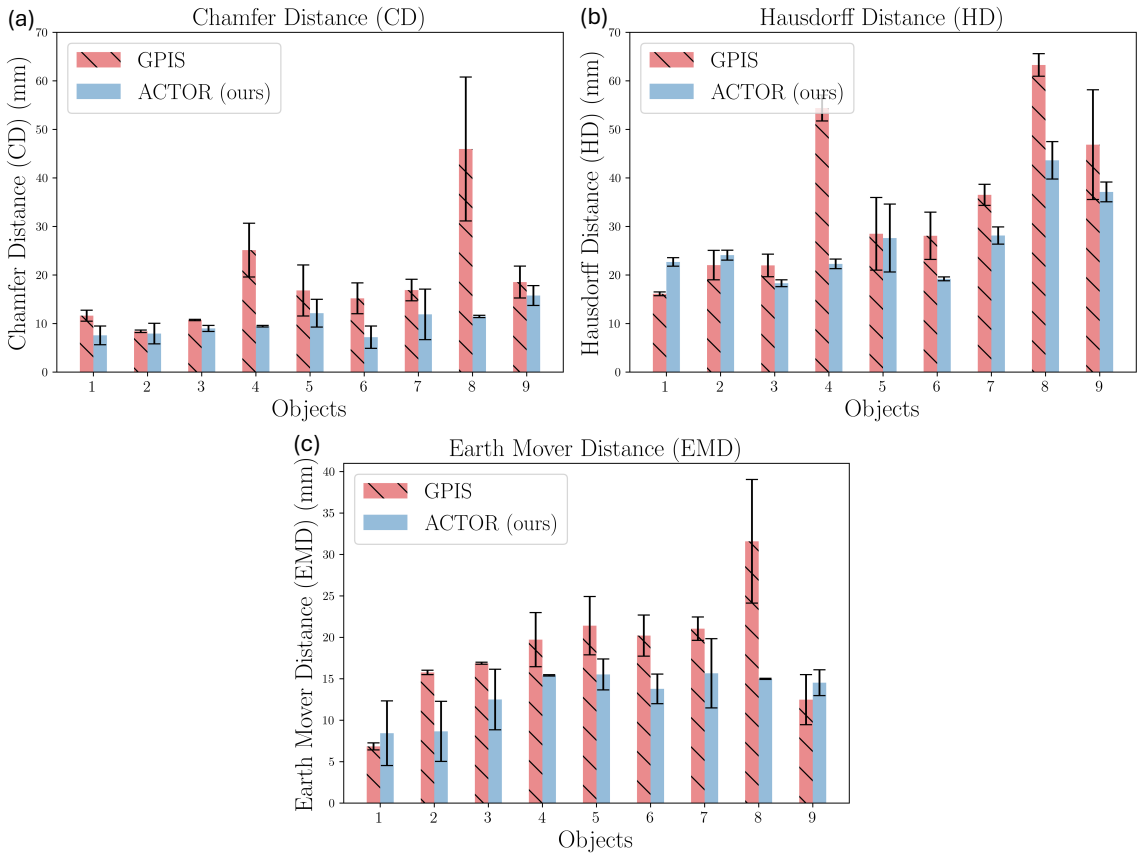


Figure 8.5: Quantitative reconstruction results. Object numbered as follows: {1: Bottle 1, 2: Bottle 2, 3: Can, 4: Detergent, 5: Cup 1, 6: Cup 2, 7: Cup 3, 8: Wineglass, 9: Spray }

Various instances of the same category were chosen to show the reconstruction capabilities of the proposed framework. The objects also have axes of symmetry which increases the pose estimation challenge.

8.3.2 Active Tactile-based Deep Self-Supervised Category-level Transparent Object Reconstruction

In order to initialise the active exploration phase for tactile data collection, a coarse bounding-box information was necessary which can be provided by the user or automatically detected using an RGB camera with off-the-shelf object detection techniques. The height of the occupancy grid was set constant for every object at 0.4m which was larger than the biggest object. While the proposed network can upsample an input point cloud with even 20 points, reconstruction with acceptable accuracy was obtained with 100 points or more as input. For each object, 10 tactile point clouds with point number between 100 and 120 points are extracted using the active exploration strategy and used for reconstruc-

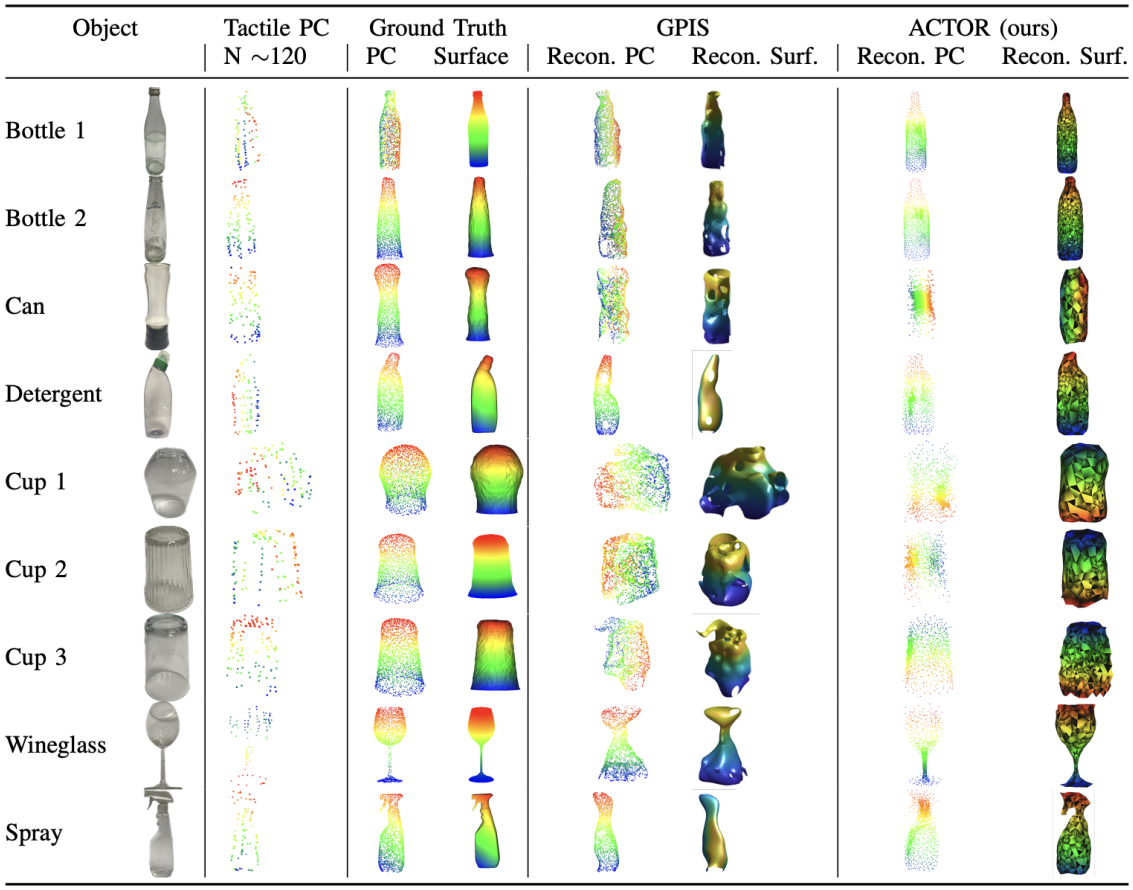


Figure 8.6: Qualitative reconstruction results of the proposed method in comparison with Gaussian process implicit surfaces for unknown real test objects.

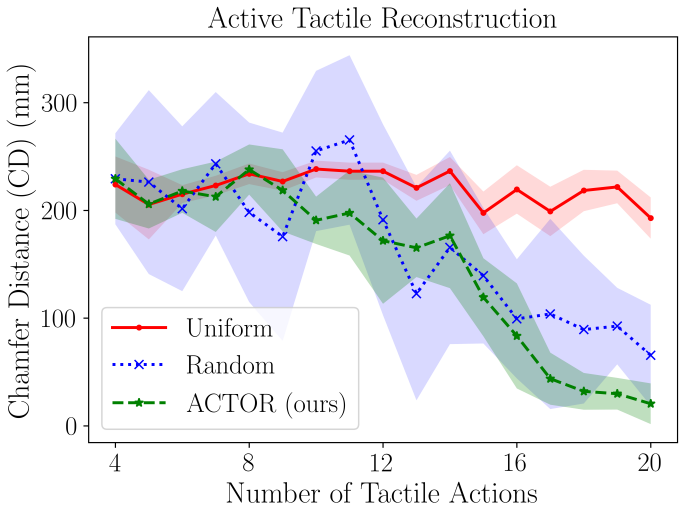


Figure 8.7: Active tactile reconstruction accuracy evaluated using the chamfer distance with ground-truth

tion. The ground-truth point cloud and mesh are obtained by spray-painting the objects and using a scanning device. For evaluation, the following performance metrics were used: Hausdorff distance, Chamfer distance and Earth Mover distance (Fei et al., 2022). Chamfer distance is described in Sec. 8.2.2. Given two points S_1 and S_2, the Hausdorff distance is defined as (Berger et al., 2013):

$$HD(S_1,S_2) = \max\{\max_{x\in S_1}\min_{y\in S_2}\{||x-y||_2\}, \max_{y\in S_2}\min_{x\in S_1}\{||y-x||_2\}\} \tag{8.6}$$

The HD represents the maximum distance between the two point sets which can be affected by extreme outliers during the reconstruction. The earth mover distance (EMD) finds a bijection $\phi : S_1 \rightarrow S_2$ to minimise the average distance between corresponding points in the point clouds as:

$$EMD(S_1,S_2) = \min_{\phi:S_1\rightarrow S_2}\frac{1}{|S_1|}\sum_{x\in S_1}||x-\phi(x)||_2 \tag{8.7}$$

A perfect reconstruction will yield $\{CD, HD, EMD\} \rightarrow 0$ and lower values signify better reconstruction.

Baseline: Gaussian Process Implicit Surfaces (GPIS) was used as baseline as it is widely deployed in the state-of-the-art methods for tactile-based object reconstruction (Dragiev et al., 2011; Yi et al., 2016; Björkman et al., 2013; Gandler et al., 2020; Martens et al., 2016; Suresh et al., 2021; Jamali et al., 2016). For implementation, the gaussian processes (GP) for machine learning toolbox (Rasmussen and Nickisch, 2010) in MATLAB was used and the Matérn kernel was used for the GP. The identical input tactile point clouds were provided to the GPIS method as to ACTOR approach.

The quantitative results of tactile-based reconstruction using the ACTOR method and baseline GPIS method are shown in Fig. 8.5 and qualitative reconstruction results are presented in Fig. 8.6. From Fig. 8.5, it can be noted that the proposed approach yields lower CD values for all objects. For HD and EMD, apart from the bottle and spray, ACTOR performs better than the baseline GPIS approach. On average, ACTOR is 45%, 23.5% and 28% lower in CD, HD and EMD values compared to baseline GPIS ($p < 0.001$ calculated Welsh's t-test). While the quantitative results focused on local point-distances between the reconstructed and ground-truth point cloud, the qualitative results in Fig. 8.6 demonstrated the differences in reconstruction accuracy at the object level. GPIS produced warped reconstructed surfaces due to the low number of tactile points. However, the proposed method, with the help of the learnt model over the category-level synthetic objects, was able to reconstruct the object to an acceptable accuracy even with sparse input data.

Active Tactile Reconstruction: A uniform object exploration and random object ex-

Table 8.1: Confusion matrix for tactile-based object recognition

True \ Predicted	Bottle	Can	Detergent	Cup	Wineglass	Spray
Bottle	0.98	0.2	0	0	0	0
Can	0.11	0.78	0.11	0	0	0
Detergent	0.08	0	0.73	0	0	0.19
Cup	0	0	0.04	0.96	0	0
Wineglass	0	0.05	0	0	0.95	0
Spray	0	0.04	0.13	0	0	0.83

ploration strategy were used as baselines which are detailed as follows: the bounding box of the object was transformed into a grid with each grid cell of size 3 cm $\times$ 3 cm (size of the sensor patch). The grid does not encode the probabilistic occupancy as in the proposed ACTOR approach. The robot explored each grid cell in a sequential manner in the uniform strategy. In contrast, for the random strategy, the robot picked a grid cell at random for exploration. In order to have an unbiased comparison between the exploration methods, a maximum of 20 actions were chosen as on average it takes 20 actions to extract at least 100 tactile points. The model inference was initiated from the fourth action onwards to have a minimum of 20 points in the tactile point cloud. The chamfer distance metric with the ground-truth after each action provided the evaluation method. As seen in Fig. 8.7, the proposed active formulation outperforms the baselines uniform and random approaches. It can be seen that the uniform strategy required a large number of tactile actions to completely explore the object for reconstruction compared to active and random strategies. The random strategy had high variance in terms of reconstruction accuracy, which was due to the stochastic nature of the exploration, while ACTOR deterministically improved reconstruction accuracy with increasing number of tactile actions.

8.3.3 Tactile-based Transparent Object Recognition

The trained recognition model was tested on tactile point clouds extracted from the objects in Fig. 8.6. The resulting confusion matrix is shown in Tab. 8.1. An overall recognition accuracy of 87.16% is achieved. While most of the categories were recognised with

Table 8.2: Comparison of the performance with and without self-attention in the network.

	HD $\downarrow$	**CD** $\downarrow$	**EMD** $\downarrow$
	All cols. (mm)		
ACTOR (without attention)	30.65	13.96	16.66
ACTOR (with attention)	27.01	10.26	13.27

high accuracy, the can object was confused with bottle and detergent due to similarity in local curvature and detergent and spray categories are confused with each other. The confusion stems from the similarities in shape, particularly if the nozzle of the spray was not explored during active tactile object exploration. The can and bottle share the same cylindrical shape features which leads to the confusion. It must be noted that the recognition network was purely trained with synthetic data without any fine-tuning with the real world tactile point clouds.

8.4 Discussion

In this chapter, a novel framework termed ACTOR, is presented for active tactile-based category-level transparent object reconstruction. By learning with only synthetic object models, ACTOR was capable of performing real-world transparent object reconstruction through sparse tactile data. ACTOR outperformed the baseline GPIS strategy by more than 20% ($p < 0.001$ Welsh's t-test) for all evaluation metrics (CD, HD and EMD measures). The baseline GPIS method was used for object reconstruction in many prior works such as (Dragiev et al., 2011; Yi et al., 2016; Björkman et al., 2013; Gandler et al., 2020; Martens et al., 2016; Suresh et al., 2021; Jamali et al., 2016). It can be seen from the qualitative reconstruction results shown in Fig. 8.6, wherein GPIS failed to capture the shape details of the object while the proposed approach captured the global and local shape accurately (see object spray and wineglass). The proposed network ACTOR implicitly learnt important feature points and was able to reconstruct the object accurately given few sparse inputs. In Tab. 8.2, the case without using self-attention in the autoencoder is shown and the performance deteriorates (by $\sim$ 10% HD metric) compared to using the self-attention layer. The proposed active exploration strategy converged faster to reconstruct the object shape, thus improving the sample efficiency compared to the baseline random and uniform exploration strategies. The active exploration strategy achieved a value of 20mm (CD metric) within 20 tactile probing actions whereas the random and uniform strategies on average achieved 65mm and 192mm (CD metric) respectively. Tactile-based recognition of the transparent objects was also performed with an average accuracy of 87%, demonstrating the versatility of the proposed framework.

Moreover, the proposed reconstruction network was not only limited to the reconstruction of tactile point clouds of transparent objects. In fact, the same methodology has been used in Chap. 5 for the reconstruction of dense visuo-tactile point clouds wherein the sparse tactile data were collected by the robot in an autonomous manner in regions of uncertainty in the visual data. In this case, the training of the network with synthetic data was subsampled to contain points in the range of 60-1024 points to account for both sparse tactile

and dense visuo-tactile point data. Beyond the predominantly controlled contexts of industrial environments, future robotic systems intended for domestic applications are required to function effectively within unstructured settings and address the challenges presented by previously unseen objects such as glassware and cutlery in household settings. The category-level object reconstruction and pose estimation methodologies formulated within this thesis have considerable potential for application in scenarios where robots are required to manipulate unknown objects. While the application demonstrated in this chapter is for reconstruction of transparent objects from sparse point cloud data, such an approach can be identically applied for upsampling other sparse point cloud data from sensors such as lidar/ radar especially in autonomous driving (Yang et al., 2024). In addition, this technique can speed-up 3D reconstruction processes prevalent in manufacturing industries by obviating the need for exhaustive viewpoints, allowing the reconstruction network to effectively fill-the-gaps in missing measurement data (Zhou et al., 2024).

Chapter 9

Conclusion and Future Work

9.1 Conclusions

Sharing vision and tactile perception is crucial for robots to accurately and robustly interact with objects in unstructured environments. This thesis had four main aims: (a) integrate visuo-tactile sensing to accurately and robust estimate the pose of objects in a possibly dense unstructured cluttered environment through shared and interactive perception; (b) develop a method for vision-to-tactile cross-modal transfer for task of object recognition in cases where vision modality is rendered unusable; (c) develop a method for reconstructing the shape of unknown objects through sparse tactile data by leveraging synthetic datasets and (d) implement the theoretical frameworks on real robotic systems and develop active perception methods to reduce the redundant data collection and improve overall system efficiency. This chapter summarises the contributions of the thesis and provides directions for future work.

Addressing the problem of weak-pairing between visual and tactile data, a novel recursive filtering technique for object pose estimation termed translation-invariant Quaternion filter (TIQF) and its globally optimal version stochastic TIQF (S-TIQF) were proposed in Chap. 4-5. The TIQF method is able handle sparse data that were extracted in a sequential manner from tactile sensors as well as dense data that were extracted in a one-shot batch-wise manner from visual sensors and efficiently and accurately handled the corresponding dense-sparse point cloud registration problem. S-TIQF improved upon TIQF by relaxing the need for a good initialisation to avoid local minina (a typical issue for locally optimal methods (Pomerleau et al., 2013)) through a stochastic initial alignment technique using an annealing procedure. Furthermore, it did not require a precise object model and also estimated the 3D scale in addition to the rotation and translation estimates for category-level reconstructed objects. Through extensive experiments across various benchmarks and

real robot experiments with objects in dense clutter, the accuracy and efficacy of both the TIQF and S-TIQF methods has been demonstrated. The proposed methods were compared against various state-of-the-art methods including popular techniques such as ICP (Besl and McKay, 1992), globally optimal robust registration methods such as TEASER++ (Yang et al., 2020) as well as learning-based approaches like PREDATOR (Huang et al., 2021a). In fact, the proposed method significantly outperformed state-of-the-art methods in terms of pose accuracy, especially in the case of sparse tactile data. The two-robot team coordinated with each other to autonomously declutter a randomly dense cluttered scene and extracted the pose of the pose of the target object. The decluttering was orchestrated by a novel scene graph approach termed declutter graph which encodes the spatial and support relationships between various objects in the scene. The visuo-tactile data was used to reconstruct the object shape using a novel joint information gain approach. The joint information gain metric reduced the number of robot actions necessary to accurately reconstruct the object shape. Moreover, S-TIQF was also deployed to rectify the hand-eye calibration error using arbitrary objects without the need for a specific calibration grid.

During interactive perception, it's likely that the target objects moves and dynamic pose tracking needs to be performed. Furthermore, objects may have intrinsic articulation that cannot be distinguished from a rigid object through visual perception alone. In Chap. 6, a novel full-fledged framework for visuo-tactile based interactive perception to detect, track and manipulate unknown objects (single, multiple or articulated objects) without assuming any prior knowledge regarding object shape or dynamics has been presented. The proposed method termed ArtReg achieved accurate pose tracking (average error < 2 cm) for all types of object (single, multiple, or articulated). Moreover, precise pose tracking was required for goal-oriented closed-loop control of these objects, and was demonstrated under various challenging conditions. These conditions include variations in the objects' centre of mass, as well as scenarios characterised by low illumination and complex backgrounds. Various types of robot actions were performed, such as pushing in the case of single or articulated objects with revolute joints and hold-pull actions involving coordination of both robots to pull apart objects with prismatic joints. The actions were executed autonomously by the robots, which discerned the type of object through iterative manipulation actions analogous to human interaction to identify the properties of the object. ArtReg was also compared with various state-of-the-art methods and outperforms them in pose tracking accuracy on a standard benchmark dataset.

In Chap. 7, the task of vision-to-tactile cross-modal recognition was tackled. A novel cross-modal object recognition method leveraging deep neural network and active perception and learning techniques was proposed. In this regard, a novel loss function termed

VTLoss was designed which minimised the domain gap between the visual and tactile domains by minimising the distribution statistics by mapping the feature vectors to a high-dimensional subspace. The proposed method in Chap. 7 outperformed the limited state-of-the-art works in this domain such as Falco et al. (2019). The presented framework also leveraged active learning with entropy formulation for vision-based object recognition and reduced the expensive cost of data annotation. Moreover, since tactile sensing is action conditioned, the sample efficiency of the tactile actions was improved using a greedy information-gain based exploration approach that takes less than 10 probing actions to recognise the object as compared with baseline exploration strategies that required more than 20 probing actions.

Similarly to object pose estimation, tactile perception can augment and refine visual perception for object reconstruction. Furthermore, for transparent and specular objects, vision sensors are ineffective for extracting 3D point cloud data. In Chap. 8, a novel framework termed ACTOR has been presented for the category-level object reconstruction of transparent objects with tactile sensing. Since collection and annotation of a large-scale tactile dataset is prohibitively expensive, synthetic data was leveraged for the training of the reconstruction network. The synthetic data was extracted from open-source object models belonging to the same categories as the real-world test objects. The neural network trained with only synthetic data was directly used for inference with real tactile point cloud data without any synthetic-to-real fine tuning. The proposed methodology demonstrates a superior reconstruction accuracy, exceeding the baseline Gaussian Process Implicit Surfaces (GPIS) method by over 20%. It is important to note that the GPIS method was often employed in the domain of object reconstruction. Furthermore, for opaque objects, tactile sensing was leveraged to reconstruct areas of uncertainty from visual sensing during shared perception.

This thesis addressed the research question stated in Chap. 1 and has presented several novel contributions for visuo-tactile-based perception and learning in robotics primarily for three application domains: object pose estimation, object recognition, and reconstruction. The theoretical frameworks developed within this thesis process data obtained from visual and tactile sensors as raw 3D point clouds. Consequently, these frameworks demonstrate wide applicability across various types of current commercial sensors and can exhibit adaptability to future advancements in sensor technologies. Furthermore, the proposed interactive perception formulations were also agnostic to the type of robot embodiment: active visual perception actions required at least 6 DoF manipulator robots, which are provided by most robot manufacturers. Active tactile perception actions were not gripper dependent, since simple probing actions can be performed even with simple point-contact

end-effectors. The algorithms proposed in this thesis can be applied in various real-world applications in manufacturing industries and household settings. The pose estimation and object reconstruction methods discussed in the thesis are essential for autonomous manipulation of objects, especially in unstructured cluttered settings. The cross-modal recognition method proves advantageous in challenging conditions, such as low-light environments or scenarios lacking direct line-of-sight, wherein robots can utilize tactile sensing to accomplish the task. Moreover, the methods developed in this thesis find applications beyond robotics and are detailed in Sec. 9.2.

9.2 Future Work

9.2.1 Extension of pose estimation formulation to other domains

The pose estimation and tracking algorithms developed in Chap. 4-6 such as S-TIQF and ArtReg are generic and depends only on 3D points, hence they can be deployed for other dense-sparse sensing modalities such as radar or lidar point-clouds in autonomous driving domain. Similarly, the active perception methodology developed is based on 3D points and occupancy grid which can also be extended to sensors beyond vision and tactile. The dynamic pose tracking of articulated objects presented in Chap. 6 involves 1 DoF prismatic and/or revolute joints. The proposed framework for tracking and manipulation of articulated objects can be extended for articulated objects with other types of joints such as universal joint, screw joints, ball joint and so on. Both S-TIQF and ArtReg algorithms work on correspondences between two point clouds, so it can readily be combined as a 'backend' with learning-based approaches as 'front-end' providing semantic segmentation or feature keypoint correspondences of measured point cloud data to improve the tracking in complex scenarios.

9.2.2 Incorporating robot uncertainties in sensor measurements

As mentioned in Chap. 5, the accuracy of the tactile sensor measurements were dependent on the kinematic accuracy of the robots. The collaborative robots used in this thesis such as Universal Robots and Franka Emika Panda have a pose repeatability of 0.1mm and empirical analysis showed the pose accuracy has a standard deviation of upto 4mm. Since the tactile sensors were rigidly attached to the robot gripper, the tactile point clouds inherently incorporate the uncertainty from the robot kinematics. Another avenue for future work involves the integration of uncertainty quantification within the registration process, which has the potential to further mitigate calibration errors. The measurement noise term

within the Kalman filter can be modified to include the noise arising from the sensor as well as the robot kinematics.

9.2.3 Extension of cross-modal perception

While this thesis focused on the shape of objects for shared visuo-tactile perception, there are other properties of the objects, such as surface texture, that can also be effectively transferred across domains. The Chap. 7 demonstrated shape recognition through vision to tactile cross-modal adaptation, the opposite i.e., transfer learning from tactile to vision perception is also an interesting problem for future work. The concept known as synaesthesia in humans is the phenomenon of stimulation of one sensory or cognitive pathway that leads to involuntary experiences in a second sensory or cognitive pathway. Emulating this for robots, for instance, based on sensed tactile signals, a neural network can generate the image of the surface which can allow the robot to interact blindly with objects. Although preliminary investigations have been conducted on this issue (Purri and Dana, 2020; Lee et al., 2019a), more comprehensive research is necessary. With the advent of large foundation models, there have been increasing number of works in literature for text and vision-based models demonstrating impressive performance across applications such as language prompting for image segmentation, object detection and so on (Kirillov et al., 2023; Zhang et al., 2024). Incorporating tactile data within vision-language-tactile models can enable seamless cross-modal transfer between modalities. This can also open the doors for new applications in human-robot interaction, for instance human can command the robot through language prompts to grab the soft toy from a cluttered bin and the robot can utilise visual and tactile perception to retrieve the required object and handover to the human. However, this necessitates large-scale labelled tactile datasets which is an open challenge due to the variations in type and outputs from tactile sensors as well as time consuming contact actions necessary for gathering tactile data. Simulation of tactile data may provide a possible avenue for tackling the problem as detailed below.

9.2.4 Improving simulation for training with tactile data

In Chap. 8, simulated data has been used to train a neural network by mimicking contact points through a sub-sampled point cloud. The network has been demonstrated to perform on real tactile data without any fine tuning for the task of shape reconstruction, thus showing the power of simulation for training large neural networks. However, the precise simulation of tactile sensors extending beyond mere point contacts, such as the accurate simulation of friction of different surfaces is a promising direction for future work. There have been recent efforts for simulating vision-based tactile data that generates the

contact 'image' (Wang et al., 2022b; Ding et al., 2020) whereas more research attention is needed to simulate the raw signals from tactile sensors arrays. Another challenge for developing a cross-platform simulator is the existence of various types of tactile sensors with differing outputs signals. Developing a common data-structure or feature-map from different types of raw signals can help in unifying the simulation setup. Many existing simulators for instance Tacto (Wang et al., 2022b) and Tactile-Gym (Church et al., 2022) are based on existing rigid-body physics simulators such as PyBullet wherein simulating soft bodies can be approximated at the cost of accuracy. Some recent works based on Finite Element Methods (FEM) such as (Narang et al., 2021; Kim et al., 2013) can model soft-body contacts more accurately. Differentiable simulations capable of handling soft-bodies will be necessary for training of large networks.

9.2.5 Improvement of hardware: robust dexterous hands with tactile sensing

In this thesis, two-finger grippers capable of antipodal grasping have been used which can be integrated with different types of tactile sensors in a straightforward manner. They also can be deployed with off-the-shelf planning algorithms for prehensile and non-prehensile manipulation. However, when manipulating complex and non-rigid objects, dexterous hands outperform the two-finger grippers (Kappassov et al., 2015). State-of-the-art dexterous hands with tactile sensing have demonstrated remarkable dexterity and functionality mimicking human hands but lack robustness, scalability and affordability (Andrychowicz et al., 2020; Eguiluz et al., 2017). This has resulted in many research groups developing their own ad-hoc created robotic hands through rapid prototyping (Bhirangi et al., 2023; Liu et al., 2022a). However, low-cost, robust, and standardised multi-fingered dexterous hands with distributed tactile sensing capable of lifting reasonable payloads (< 10 kg) are critical for the manipulation of complex objects in real-world unstructured environments. Moreover, the developed methodologies in this thesis are based only on contact positions and forces and are agnostic to type of robot embodiment. Thus, they can readily be deployed on current and future multi-fingered dexterous hands.

9.2.6 Extension of visuo-tactile perception beyond robotics

Beyond robot manipulation, visuo-tactile perception finds applications in other fields as well. A detailed review of the state-of-the-art for technologies and algorithms for in-vehicle interaction has been conducted during this doctoral study but not presented in this thesis and is available in Murali et al. (2022c). Many contact surfaces within vehicles can be sensorised and the tactile data may be combined with in-vehicle camera data to enrich

the information regarding the passenger state. It can be useful for applications such as human pose tracking and human state monitoring within the vehicle which can enable other use-cases such as augmented reality/ virtual reality, fatigue detection, ergonomic posture recommendation, and so on (Murali et al., 2022c).

In conclusion, this thesis has introduced novel methodologies for visuo-tactile perception and learning in robotics, with a particular emphasis on object pose estimation, recognition, and reconstruction. The findings demonstrate that the integration of visual and tactile sensory data allows robots to accurately determine the pose of objects in unstructured and cluttered environments, even when dealing with objects featuring shiny or transparent surfaces. Additionally, this work has shown that tactile sensing proves effective for surface reconstruction of transparent objects when combined with novel deep learning techniques for point cloud upsampling. Furthermore, a novel framework for vision-to-tactile cross-modal transfer learning method has been proposed, enabling robotic systems to switch to tactile sensing in scenarios where vision is compromised, thus enhancing system robustness. The methods presented herein have the potential to facilitate a range of real-world applications for robots operating in unstructured environments and offer promising directions for future research on interactive and shared multi-modal perception in robotics.

References

Zineb Abderrahmane, Gowrishankar Ganesh, André Crosnier, and Andrea Cherubini. Visuo-tactile recognition of daily-life objects never seen or touched before. In *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 1765–1770. IEEE, 2018.

Gabriel Agamennoni, Simone Fontana, Roland Y Siegwart, and Domenico G Sorrenti. Point clouds registration with probabilistic data association. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4092–4098. IEEE, 2016.

Pulkit Agrawal, Ashvin Nair, Pieter Abbeel, Jitendra Malik, and Sergey Levine. Learning to poke by poking: Experiential learning of intuitive physics. *arXiv preprint arXiv:1606.07419*, 2016.

Hassan Alirezaei, Akihiko Nagakubo, and Yasuo Kuniyoshi. A tactile distribution sensor which enables stable measurement under high and dynamic stretch. In *2009 IEEE Symposium on 3D User Interfaces*, pages 87–93. IEEE, 2009.

PK Allen and P Michelman. Acquisition and interpretation of 3-d sensor data from touch. *IEEE Transactions on Robotics and Automation*, 6(4):397–404, 1990.

Alex Alspach, Kunimatsu Hashimoto, Naveen Kuppuswamy, and Russ Tedrake. Soft-bubble: A highly compliant dense geometry tactile sensor for robot manipulation. In *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, pages 597–604. IEEE, 2019.

David Álvarez, Máximo A Roa, and Luis Moreno. Visual and tactile fusion for estimating the pose of a grasped object. In *Iberian Robotics conference*, pages 184–198. Springer, 2019.

Shigeru Ando and Hiroyuki Shinoda. Ultrasonic emission tactile sensing. *IEEE Control Systems Magazine*, 15(1):61–69, 1995.

Alexander Andreopoulos and John K Tsotsos. 50 years of object recognition: Directions forward. *Computer vision and image understanding*, 117(8):827–891, 2013.

OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, and Alex Ray. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020.

ATI. F/t sensor: Nano17, 2025. URL `https://www.ati-ia.com/products/ft/ft_models.aspx?id=Nano17`.

Paul Bach-y Rita and Stephen W Kercel. Sensory substitution and the human–machine interface. *Trends in cognitive sciences*, 7(12):541–546, 2003.

Maria Bauza, Francois R Hogan, and Alberto Rodriguez. A data-efficient approach to precise and controlled pushing. In *Conference on Robot Learning*, pages 336–345. PMLR, 2018.

William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9368–9377, 2018.

Matthew Berger, Joshua A Levine, Luis Gustavo Nonato, Gabriel Taubin, and Claudio T Silva. A benchmark for surface reconstruction. *ACM Transactions on Graphics (TOG)*, 32(2):1–17, 2013.

Fausto Bernardini, Joshua Mittleman, Holly Rushmeier, Cláudio Silva, and Gabriel Taubin. The ball-pivoting algorithm for surface reconstruction. *IEEE transactions on visualization and computer graphics*, 5(4):349–359, 1999.

Nicolai A Bernstein. The co-ordination and regulation of movements, 1967.

Dimitris Bertsimas and John Tsitsiklis. Simulated annealing. *Statistical science*, 8(1): 10–15, 1993.

Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pages 586–606. Spie, 1992.

Tapomayukh Bhattacharjee, Ashwin A Shenoi, Daehyung Park, James M Rehg, and Charles C Kemp. Combining tactile sensing and vision for rapid haptic mapping. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1200–1207. IEEE, 2015.

Raunaq Bhirangi, Abigail DeFranco, Jacob Adkins, Carmel Majidi, Abhinav Gupta, Tess Hellebrekers, and Vikash Kumar. All the feels: A dexterous hand with large-area tactile sensing. *IEEE Robotics and Automation Letters*, 2023.

Alexander Bierbaum, Kai Welke, D Burger, Tamim Asfour, and Rüdiger Dillmann. Haptic exploration for 3d shape reconstruction using five-finger hands. In *2007 7th IEEE-RAS International Conference on Humanoid Robots*, pages 616–621. IEEE, 2007.

Joao Bimbo, Petar Kormushev, Kaspar Althoefer, and Hongbin Liu. Global estimation of an object's pose using tactile sensing. *Advanced Robotics*, 29(5):363–374, 2015.

Joao Bimbo, Shan Luo, Kaspar Althoefer, and Hongbin Liu. In-hand object pose estimation using covariance-based tactile to geometry matching. *IEEE Robotics and Automation Letters*, 1(1):570–577, 2016.

Marten Björkman, Yasemin Bekiroglu, Virgile Högman, and Danica Kragic. Enhancing visual perception of shape through tactile glances. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3180–3186. IEEE, 2013.

Jeannette Bohg, Antonio Morales, Tamim Asfour, and Danica Kragic. Data-driven grasp synthesis—a survey. *IEEE Transactions on Robotics*, 30(2):289–309, 2013.

Jeannette Bohg, Karol Hausman, Bharath Sankaran, Oliver Brock, Danica Kragic, Stefan Schaal, and Gaurav S Sukhatme. Interactive perception: Leveraging action in perception and perception in action. *IEEE Transactions on Robotics*, 33(6):1273–1291, 2017.

Gunilla Borgefors. Distance transformations in digital images. *Comp. vis., grap., and im. proc.*, 1986.

Karsten M Borgwardt, Arthur Gretton, Malte J Rasch, Hans-Peter Kriegel, Bernhard Schölkopf, and Alex J Smola. Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22(14):e49–e57, 2006.

Sofien Bouaziz, Andrea Tagliasacchi, and Mark Pauly. Sparse iterative closest point. In *Computer graphics forum*, volume 32, pages 113–123. Wiley Online Library, 2013.

Frederic Bourgault, Alexei A Makarenko, Stefan B Williams, Ben Grocholsky, and Hugh F Durrant-Whyte. Information based adaptive robotic exploration. In *IEEE/RSJ international conference on intelligent robots and systems*, volume 1, pages 540–545. IEEE, 2002.

TG Bower. *Development in infancy.* WH Freeman, 1974.

TGR Bower. Object perception in infants. *Perception*, 1(1):15–30, 1972.

Gary Bradski and Adrian Kaehler. Opencv. *Dr. Dobb's journal of software tools*, 3(2), 2000.

Martin Brossard, Axel Barrau, and Silvere Bonnabel. A code for unscented kalman filtering on manifolds (ukf-m). In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5701–5708. IEEE, 2020.

Gereon H Büscher, Risto Kõiva, Carsten Schürmann, Robert Haschke, and Helge J Ritter. Flexible and stretchable fabric-based tactile sensor. *Robotics and Autonomous Systems*, 63:244–252, 2015.

Emily W Bushnell. The decline of visually guided reaching during infancy. *Infant Behavior and Development*, 8(2):139–155, 1985.

James C. Craig, Jayne M. Kisner. Factors affecting tactile spatial acuity. *Somatosensory & motor research*, 15(1):29–45, 1998.

Raúl Cabido, David Concha, Juan José Pantrigo, and Antonio S Montemayor. High speed articulated object tracking using gpus: A particle filter approach. In *2009 10th International Symposium on Pervasive Systems, Algorithms, and Networks*, pages 757–762. IEEE, 2009.

Roberto Calandra, Andrew Owens, Dinesh Jayaraman, Justin Lin, Wenzhen Yuan, Jitendra Malik, Edward H Adelson, and Sergey Levine. More than a feeling: Learning to grasp and regrasp using vision and touch. *IEEE Robotics and Automation Letters*, 3(4):3300–3307, 2018.

Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, and Hao Su. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

Yevgen Chebotar, Karol Hausman, Zhe Su, Gaurav S Sukhatme, and Stefan Schaal. Self-supervised regrasping using spatio-temporal tactile features and reinforcement learning. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1960–1966. IEEE, 2016.

Dengsheng Chen, Jun Li, Zheng Wang, and Kai Xu. Learning canonical shape space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11973–11982, 2020.

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

Yu Cheng, Dan Wu, Saifei Hao, Yang Jie, Xia Cao, Ning Wang, and Zhong Lin Wang. Highly stretchable triboelectric tactile sensor for electronic skin. *Nano Energy*, 64: 103907, 2019.

Sachin Chitta. Moveit!: an introduction. *Robot Operating System (ROS) The Complete Reference (Volume 1)*, pages 3–27, 2016.

Daniel Choukroun, Itzhack Y Bar-Itzhack, and Yaakov Oshman. Novel quaternion kalman filter. *IEEE Transactions on Aerospace and Electronic Systems*, 42(1):174–190, 2006.

Alex Church, John Lloyd, and Nathan Lepora. Tactile sim-to-real policy transfer via real-to-sim image translation. In *Conference on Robot Learning*, pages 1645–1654. PMLR, 2022.

Cristina Garcia Cifuentes, Jan Issac, Manuel Wüthrich, Stefan Schaal, and Jeannette Bohg. Probabilistic articulated real-time tracking for robot manipulation. *IEEE Robotics and Automation Letters*, 2(2):577–584, 2016.

Ignasi Clavera, David Held, and Pieter Abbeel. Policy transfer via modularity and reward guiding. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1537–1544. IEEE, 2017.

Cl Connolly. The determination of next best views. In *Proceedings. 1985 IEEE international conference on robotics and automation*, volume 2, pages 432–435. IEEE, 1985.

Tadeo Corradi, Peter Hall, and Pejman Iravani. Object recognition combining vision and touch. *Robotics and biomimetics*, 4:1–10, 2017.

Erwin Coumans and Yunfei Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. `http://pybullet.org`, 2016.

Shaowei Cui, Rui Wang, Junhang Wei, Jingyi Hu, and Shuo Wang. Self-attention based visual-tactile fusion learning for predicting grasp outcomes. *IEEE Robotics and Automation Letters*, 5(4):5827–5834, 2020a.

Shaowei Cui, Rui Wang, Junhang Wei, Fanrong Li, and Shuo Wang. Grasp state assessment of deformable objects using visual-tactile fusion perception. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 538–544. IEEE, 2020b.

Mark R Cutkosky and William Provancher. Force and tactile sensing. In *Springer Handbook of Robotics*, pages 717–736. Springer, 2016.

Ravinder Dahiya. E-skin: from humanoids to humans. *Proc. of the IEEE*, 107(2):247–252, 2019.

Ravinder Dahiya, Nivasan Yogeswaran, Fengyuan Liu, Libu Manjakkal, Etienne Burdet, Vincent Hayward, and Henrik Jörntell. Large-area soft e-skin: The challenges beyond sensor designs. *Proceedings of the IEEE*, 107(10):2016–2033, 2019.

Ravinder S Dahiya and Maurizio Valle. *Robotic tactile sensing: technologies and system*. Springer Science & Business Media, 2012.

Ravinder S Dahiya, Giorgio Metta, Maurizio Valle, and Giulio Sandini. Tactile sensing—from humans to humanoids. *IEEE transactions on robotics*, 26(1):1–20, 2009.

Michael Danielczuk, Andrey Kurenkov, Ashwin Balakrishna, Matthew Matl, David Wang, Roberto Martín-Martín, Animesh Garg, Silvio Savarese, and Ken Goldberg. Mechanical search: Multi-step retrieval of a target object occluded by clutter. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 1614–1621. IEEE, 2019.

Jeffrey Delmerico, Stefan Isler, Reza Sabzevari, and Davide Scaramuzza. A comparison of volumetric information gain metrics for active 3d object reconstruction. *Autonomous Robots*, 42(2):197–208, 2018.

Haowen Deng, Tolga Birdal, and Slobodan Ilic. Ppfnet: Global context aware local features for robust 3d point matching. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 195–205, 2018.

Xinke Deng, Junyi Geng, Timothy Bretl, Yu Xiang, and Dieter Fox. icaps: Iterative category-level object pose and shape estimation. *IEEE Robotics and Automation Letters*, 7(2):1784–1791, 2022.

Jérémie Deray and Joan Solà. Manif: A micro Lie theory library for state estimation in robotics applications. *Journal of Open Source Software*, 5(46):1371, 2020. doi: 10.21105/joss.01371. URL https://doi.org/10.21105/joss.01371.

Snehal Dikhale, Karankumar Patel, Daksh Dhingra, Itoshi Naramura, Akinobu Hayashi, Soshi Iba, and Nawid Jamali. Visuotactile 6d pose estimation of an in-hand object using vision and tactile sensor data. *IEEE Robotics and Automation Letters*, 7(2):2148–2155, 2022.

Zihan Ding, Nathan F Lepora, and Edward Johns. Sim-to-real transfer for optical tactile sensing. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1639–1645. IEEE, 2020.

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

Alin Drimus, Gert Kootstra, Arne Bilberg, and Danica Kragic. Design of a flexible tactile sensor for classification of rigid and deformable objects. *Robotics and Autonomous Systems*, 62(1):3–15, 2014.

John Duchi. Derivations for linear algebra and optimization. *Berkeley, California*, 3(1): 2325–5870, 2007.

A Gomez Eguiluz, Iñaki Rañó, Sonya A Coleman, and T Martin McGinnity. Reliable object handover through tactile force sensing and effort control in the shadow robot hand. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 372–377. IEEE, 2017.

Ben Eisner, Harry Zhang, and David Held. Flowbot3d: Learning 3d articulation flow to manipulate articulated objects. *arXiv preprint arXiv:2205.04382*, 2022.

Marc O Ernst and Martin S Banks. Humans integrate visual and haptic information in a statistically optimal fashion. *Nature*, 2002.

Pablo Escobedo, Markellos Ntagios, Dhayalan Shakthivel, William T Navaraj, and Ravinder Dahiya. Energy generating electronic skin with intrinsic tactile sensing without touch sensors. *IEEE Transactions on Robotics*, 37(2):683–690, 2020.

Pietro Falco, Shuang Lu, Ciro Natale, Salvatore Pirozzi, and Dongheui Lee. A transfer learning approach to cross-modal object recognition: from visual observation to robotic haptic exploration. *IEEE Transactions on Robotics*, 35(4):987–998, 2019.

Nima Fazeli, Miquel Oller, Jiajun Wu, Zheng Wu, Joshua B Tenenbaum, and Alberto Rodriguez. See, feel, act: Hierarchical learning for complex manipulation skills with multisensory fusion. *Science Robotics*, 4(26), 2019.

Ben Fei, Weidong Yang, Wen-Ming Chen, Zhijun Li, Yikang Li, Tao Ma, Xing Hu, and Lipeng Ma. Comprehensive review of deep learning-based 3d point cloud completion processing and analysis. *IEEE Transactions on Intelligent Transportation Systems*, 2022.

Di Feng, Xiao Wei, Lars Rosenbaum, Atsuto Maki, and Klaus Dietmayer. Deep active learning for efficient training of a lidar 3d object detector. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 667–674. IEEE, 2019.

Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2786–2793. IEEE, 2017.

Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

Jeremy A Fishel and Gerald E Loeb. Sensing tactile microvibrations with the biotac—comparison with human sensitivity. In *2012 4th IEEE RAS & EMBS international conference on biomedical robotics and biomechatronics (BioRob)*, pages 1122–1127. IEEE, 2012.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

Gabriela Zarzar Gandler, Carl Henrik Ek, Mårten Björkman, Rustam Stolkin, and Yasemin Bekiroglu. Object shape estimation and modeling, based on sparse gaussian process implicit surfaces, combining visual data and tactile exploration. *Robotics and Autonomous Systems*, 2020.

Wei Gao and Russ Tedrake. Filterreg: Robust and efficient probabilistic point-set registration using gaussian filter and twist parameterization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11095–11104, 2019.

Liuhao Ge, Zhou Ren, Yuncheng Li, Zehao Xue, Yingying Wang, Jianfei Cai, and Junsong Yuan. 3d hand shape and pose estimation from a single rgb image. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 10833–10842, 2019.

Michael Gentner, Prajval Kumar Murali, and Mohsen Kaboli. Gmcr: Graph-based maximum consensus estimation for point cloud registration. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4967–4974. IEEE, 2023.

Daniel Glozman, Moshe Shoham, and Anath Fischer. A surface-matching technique for robot-assisted registration. *Computer Aided Surgery*, 6(5):259–269, 2001.

Dirk Goger, Nicolas Gorges, and Heinz Worn. Tactile sensing for an anthropomorphic robotic hand: Hardware and signal processing. In *2009 IEEE International Conference on Robotics and Automation*, pages 895–901. IEEE, 2009.

Christophe Gonzales and Séverine Dubuisson. Combinatorial resampling particle filter: An effective and efficient method for articulated object tracking. *International Journal of Computer Vision*, 112:255–284, 2015.

K"Gopal" Gopalakrishnan and Ken Goldberg. D-space and deform closure grasps of deformable parts. *The International Journal of Robotics Research*, 24(11):899–910, 2005.

Monica Gori, Michela Del Viva, Giulio Sandini, and David C Burr. Young children do not integrate visual and haptic form information. *Current Biology*, 18(9):694–698, 2008.

Vincent Granville, Mirko Krivánek, and J-P Rasson. Simulated annealing: A proof of convergence. *IEEE transactions on pattern analysis and machine intelligence*, 16(6): 652–656, 1994.

Arthur Gretton, Dino Sejdinovic, Heiko Strathmann, Sivaraman Balakrishnan, Massimiliano Pontil, Kenji Fukumizu, and Bharath K Sriperumbudur. Optimal kernel choice for large-scale two-sample tests. In *Advances in neural information processing systems*, pages 1205–1213. Citeseer, 2012.

Enrico Grosso, Giulio Sandini, and Massimo Tistarelli. 3d object reconstruction using stereo and motion. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1465–1476, 1989.

Xian-Feng Han, Hamid Laga, and Mohammed Bennamoun. Image-based 3d object reconstruction: State-of-the-art and trends in the deep learning era. *IEEE transactions on pattern analysis and machine intelligence*, 43(5):1578–1604, 2019.

Yvette Hatwell. Motor and cognitive functions of the hand in infancy and childhood. *International Journal of Behavioral Development*, 10(4):509–526, 1987.

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

Paul Hebert, Thomas Howard, Nicolas Hudson, Jeremy Ma, and Joel W Burdick. The next best touch for model-based localization. In *2013 IEEE International Conference on Robotics and Automation*, pages 99–106, 2013.

Alan Hein and Richard Held. Dissociation of the visual placing response into elicited and guided components. *Science*, 158(3799):390–392, 1967.

Hannah B Helbig and Marc O Ernst. Optimal integration of shape information from vision and touch. *Experimental brain research*, 179:595–606, 2007.

Richard Held and Joseph A Bauer. Visually guided reaching in infant monkeys after restricted rearing. *Science*, 155(3763):718–720, 1967.

Richard Held and Alan Hein. Movement-produced stimulation in the development of visually guided behavior. *Journal of comparative and physiological psychology*, 56(5):872, 1963.

Randall Blake Hellman. *Haptic perception, decision-making, and learning for manipulation with artificial hands*. Arizona State University, 2016.

Darrall Henderson, Sheldon H Jacobson, and Alan W Johnson. The theory and practice of simulated annealing. *Handbook of metaheuristics*, pages 287–319, 2003.

James M Hillis, Marc O Ernst, Martin S Banks, and Michael S Landy. Combining sensory information: mandatory fusion within, but not between, senses. *Science*, 298(5598): 1627–1630, 2002.

Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Computer Vision–ACCV 2012: 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9, 2012, Revised Selected Papers, Part I 11*, pages 548–562. Springer, 2013.

Radu Horaud, Miles Hansard, Georgios Evangelidis, and Clément Ménier. An overview of depth cameras and range scanners based on time-of-flight technologies. *Machine vision and applications*, 27(7):1005–1020, 2016.

Berthold KP Horn. Closed-form solution of absolute orientation using unit quaternions. *Josa a*, 4(4):629–642, 1987.

Armin Hornung, Kai M Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. Octomap: An efficient probabilistic 3d mapping framework based on octrees. *Autonomous robots*, 34:189–206, 2013.

Matanya B Horowitz and Joel W Burdick. Combined grasp and manipulation planning as a trajectory optimization problem. In *2012 IEEE International Conference on Robotics and Automation*, pages 584–591. IEEE, 2012.

Kaijen Hsiao, L Kaelbling, and Tomás Lozano-Pérez. Task-driven tactile exploration. *Robotics: Science and Systems VI*, 2010.

Kaijen Hsiao, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Robust grasping under object pose uncertainty. *Autonomous Robots*, 31(2):253–268, 2011.

Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Andreas Wieser, and Konrad Schindler. Predator: Registration of 3d point clouds with low overlap. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 4267–4276, 2021a.

Xiaoshui Huang, Guofeng Mei, Jian Zhang, and Rana Abbas. A comprehensive survey on point cloud registration. *arXiv preprint arXiv:2103.02690*, 2021b.

Ivo Ihrke, Kiriakos N Kutulakos, Hendrik PA Lensch, Marcus Magnor, and Wolfgang Heidrich. Transparent and specular object reconstruction. In *Computer graphics forum*, volume 29, pages 2400–2426. Wiley Online Library, 2010.

Nawid Jamali, Carlo Ciliberto, Lorenzo Rosasco, and Lorenzo Natale. Active perception: Building objects' models using tactile exploration. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 179–185. IEEE, 2016.

Carlos A Jara, Jorge Pomares, Francisco A Candelas, and Fernando Torres. Control framework for dexterous manipulation using dynamic visual servoing and tactile sensors' feedback. *Sensors*, 14(1):1787–1804, 2014.

AH Jazwinski. Stochastic processes and filtering theory. 1970.

Marc Jeannerod. The timing of natural prehension movements. *Journal of motor behavior*, 16(3):235–254, 1984.

Yan-Bin Jia and Michael Erdmann. Pose and motion from contact. *The International Journal of Robotics Research*, 18(5):466–487, 1999.

Yan-Bin Jia and Jiang Tian. Surface patch reconstruction from "one-dimensional" tactile data. *IEEE Transactions on Automation Science and Engineering*, 7(2):400–407, 2009.

Jiaqi Jiang, Guanqun Cao, Jiankang Deng, Thanh-Toan Do, and Shan Luo. Robotic perception of transparent objects: A review. *IEEE Transactions on Artificial Intelligence*, 2023.

Liang-Ting Jiang and Joshua R Smith. Seashell effect pretouch sensing for robotic grasping. In *2012 IEEE International Conference on Robotics and Automation*, pages 2851–2858. IEEE, 2012.

Minghe Jin, Haiwei Gu, Shaowei Fan, Yuanfei Zhang, and Hong Liu. Object shape recognition approach for sparse point clouds from tactile exploration. In *2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 558–562. IEEE, 2013.

Roland S Johansson and Ake B Vallbo. Tactile sensibility in the human hand: relative and absolute densities of four types of mechanoreceptive units in glabrous skin. *The Journal of physiology*, 286(1):283–300, 1979.

Roland S Johansson and Goran Westling. Roles of glabrous skin receptors and sensorimotor memory in automatic control of precision grip when lifting rougher or more slippery objects. *Experimental brain research*, 56:550–564, 1984.

Edward Johns, Stefan Leutenegger, and Andrew J Davison. Deep learning a grasp function for grasping under gripper pose uncertainty. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4461–4468. IEEE, 2016.

Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3668–3678, 2015.

Micah K Johnson, Forrester Cole, Alvin Raj, and Edward H Adelson. Microgeometry capture using an elastomeric sensor. *ACM Transactions on Graphics (TOG)*, 30(4):1–8, 2011.

Minhyun Jung, Jumi Lee, Sujaya Kumar Vishwanath, Oh-Sun Kwon, Chi Won Ahn, Kwanwoo Shin, and Sanghun Jeon. Flexible multimodal sensor inspired by human skin based on hair-type flow, temperature, and pressure. *Flexible and Printed Electronics*, 5(2): 025003, 2020.

Mohsen Kaboli and Gordon Cheng. Robust tactile descriptors for discriminating objects from textural properties via artificial robotic skin. *IEEE Transactions on Robotics*, 34 (4):985–1003, 2018.

Mohsen Kaboli, Kunpeng Yao, and Gordon Cheng. Tactile-based manipulation of deformable objects with dynamic center of mass. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 752–757. IEEE, 2016.

Mohsen Kaboli, Di Feng, Kunpeng Yao, Pablo Lanillos, and Gordon Cheng. A tactile-based framework for active object learning and discrimination using multimodal robotic skin. *IEEE Robotics and Automation Letters*, 2(4):2143–2150, 2017.

Mohsen Kaboli, Di Feng, and Gordon Cheng. Active tactile transfer learning for object discrimination in an unstructured environment using multimodal robotic skin. *International Journal of Humanoid Robotics*, 15(01):1850001, 2018.

Mohsen Kaboli, Kunpeng Yao, Di Feng, and Gordon Cheng. Tactile-based active object discrimination and target object search in an unknown workspace. *Autonomous Robots*, 43:123–152, 2019.

Ishay Kamon, Tamar Flash, and Shimon Edelman. Learning to grasp using visual information. In *Proceedings of IEEE International Conference on Robotics and Automation*, volume 3, pages 2470–2476. IEEE, 1996.

Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7122–7131, 2018.

Zhanat Kappassov, Juan-Antonio Corrales, and Véronique Perdereau. Tactile sensing in dexterous robot hands. *Robotics and Autonomous Systems*, 74:195–220, 2015.

Daniel Kappler, Jeannette Bohg, and Stefan Schaal. Leveraging big data for grasp planning. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4304–4311. IEEE, 2015.

Rainer Kartmann, Fabian Paus, Markus Grotz, and Tamim Asfour. Extraction of physically plausible support relations to predict and validate manipulation action effects. *IEEE Robotics and Automation Letters*, 3(4):3991–3998, 2018.

Yo Kato, Toshiharu Mukai, Tomonori Hayakawa, and Tetsuyoshi Shibata. Tactile sensor without wire and sensing element in the tactile region based on eit method. In *Sensors, 2007 Ieee*, pages 792–795. IEEE, 2007.

Dov Katz and Oliver Brock. Manipulating articulated objects with interactive perception. In *2008 IEEE International Conference on Robotics and Automation*, pages 272–277. IEEE, 2008.

Moslem Kazemi, Jean-Sebastien Valois, J Andrew Bagnell, and Nancy Pollard. Robust object grasping using force compliant motion primitives. In *Proceedings of the Robotics: Science and Systems Conference (RSS)*, pages 177–185, 2012.

Kenta-Tanaka. probreg, 2019. URL `https://probreg.readthedocs.io/en/latest/`.

Heba Khamis, Raquel Izquierdo Albero, Matteo Salerno, Ahmad Shah Idil, Andrew Loizou, and Stephen J Redmond. Papillarray: An incipient slip sensor for dexterous robotic or prosthetic manipulation–design and prototype validation. *Sensors and Actuators A: Physical*, 270:195–204, 2018.

Heba Khamis, Benjamin Xia, and Stephen J Redmond. A novel optical 3d force and displacement sensor–towards instrumenting the papillarray tactile sensor. *Sensors and Actuators A: Physical*, 291:174–187, 2019.

Junggon Kim, Kunihiro Iwamoto, James J Kuffner, Yasuhiro Ota, and Nancy S Pollard. Physically based grasp quality evaluation under pose uncertainty. *IEEE Transactions on Robotics*, 29(6):1424–1439, 2013.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023.

Konrad P Körding and Daniel M Wolpert. Bayesian integration in sensorimotor learning. *Nature*, 427(6971):244–247, 2004.

Danica Kragic, S Crinier, Dietrich Brunn, and Henrik I Christensen. Vision and tactile sensing for real world tasks. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, volume 1, pages 1545–1550. IEEE, 2003.

Oliver Kroemer, Christoph H Lampert, and Jan Peters. Learning dynamic tactile sensing with robust vision-based training. *IEEE transactions on robotics*, 27(3):545–557, 2011.

Kenta Kumagai and Kazuhiro Shimonomura. Event-based tactile image sensor for detecting spatio-temporal fast phenomena in contacts. In *2019 IEEE World Haptics Conference (WHC)*, pages 343–348. IEEE, 2019.

Visak Kumar, Tucker Hermans, Dieter Fox, Stan Birchfield, and Jonathan Tremblay. Contextual reinforcement learning of visuo-tactile multi-fingered grasping policies. *arXiv preprint arXiv:1911.09233*, 2019.

Naveen Kuppuswamy, Alejandro Castro, Calder Phillips-Grafflin, Alex Alspach, and Russ Tedrake. Fast model-based contact patch and pose estimation for highly deformable dense-geometry tactile sensors. *IEEE Robotics and Automation Letters*, 5(2):1811–1818, 2019.

Yann Labbé, Justin Carpentier, Mathieu Aubry, and Josef Sivic. Single-view robot pose and joint angle estimation via render & compare. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1654–1663, 2021.

Mikko Lauri, Joni Pajarinen, Jan Peters, and Simone Frintrop. Multi-sensor next-best-view planning as matroid-constrained submodular maximization. *IEEE Robotics and Automation Letters*, 5(4):5323–5330, 2020.

Susan J Lederman and Roberta L Klatzky. Hand movements: A window into haptic object recognition. *Cognitive psychology*, 19(3):342–368, 1987.

Hyung-Kew Lee, Jaehoon Chung, Sun-Il Chang, and Euisik Yoon. Normal and shear force measurement using a flexible polymer tactile sensor with embedded multiple capacitors. *Journal of Microelectromechanical Systems*, 17(4):934–942, 2008.

Jet-Tsyn Lee, Danushka Bollegala, and Shan Luo. "touching to see" and "seeing to feel": Robotic cross-modal sensory data generation for visual-tactile perception. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 4276–4282. IEEE, 2019a.

Mark H Lee and Howard R Nicholls. Review article tactile sensing for mechatronics—a state of the art survey. *Mechatronics*, 9(1):1–31, 1999.

Michelle A Lee, Yuke Zhu, Krishnan Srinivasan, Parth Shah, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Jeannette 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)*, pages 8943–8950. IEEE, 2019b.

Michelle A Lee, Matthew Tan, Yuke Zhu, and Jeannette Bohg. Detect, reject, correct: Crossmodal compensation of corrupted sensors. *arXiv preprint arXiv:2012.00201*, 2020.

Taeyeop Lee, Byeong-Uk Lee, Myungchul Kim, and In So Kweon. Category-level metric scale object shape and pose estimation. *IEEE Robotics and Automation Letters*, 6(4): 8575–8582, 2021.

Ian Lenz, Honglak Lee, and Ashutosh Saxena. Deep learning for detecting robotic grasps. *The International Journal of Robotics Research*, 34(4-5):705–724, 2015.

Nathan F Lepora and Benjamin Ward-Cherrier. Superresolution with an optical tactile sensor. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2686–2691. IEEE, 2015.

Sergey Levine, Peter Pastor, Alex Krizhevsky, Julian Ibarz, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International Journal of Robotics Research*, 37(4-5):421–436, 2018.

Marc Levoy, J Gerth, Brian Curless, and K Pull. The stanford 3d scanning repository. *URL http://www-graphics. stanford. edu/data/3dscanrep*, 5(10), 2005.

Jue Kun Li, Wee Sun Lee, and David Hsu. Push-net: Deep planar pushing for objects with unknown physical properties. In *Robotics: Science and Systems*, volume 14, pages 1–9, 2018.

Qiang Li, Robert Haschke, and Helge Ritter. A visuo-tactile control framework for manipulation and exploration of unknown objects. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 610–615. IEEE, 2015.

Qiang Li, Oliver Kroemer, Zhe Su, Filipe Fernandes Veiga, Mohsen Kaboli, and Helge Joachim Ritter. A review of tactile information: Perception and action through touch. *IEEE Transactions on Robotics*, 36(6):1619–1634, 2020a.

Rui Li, Robert Platt, Wenzhen Yuan, Andreas ten Pas, Nathan Roscup, Mandayam A Srinivasan, and Edward Adelson. Localization and manipulation of small parts using gelsight tactile sensing. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3988–3993. IEEE, 2014.

Xiaolong Li, He Wang, Li Yi, Leonidas J Guibas, A Lynn Abbott, and Shuran Song. Category-level articulated object pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3706–3715, 2020b.

Fengyuan Liu, Sweety Deswal, Adamos Christou, Mahdieh Shojaei Baghini, Radu Chirila, Dhayalan Shakthivel, Moupali Chakraborty, and Ravinder Dahiya. Printed synaptic transistor–based electronic skin for robots to feel and learn. *Science Robotics*, 7(67): eabl7286, 2022a.

Hongbin Liu, Xiaojing Song, Thrishantha Nanayakkara, Lakmal D Seneviratne, and Kaspar Althoefer. A computationally fast algorithm for local contact shape and pose classification using a tactile array sensor. In *2012 IEEE International Conference on Robotics and Automation*, pages 1410–1415. IEEE, 2012.

Huaping Liu, Di Guo, and Fuchun Sun. Object recognition using tactile measurements: Kernel sparse coding methods. *IEEE Transactions on Instrumentation and Measurement*, 65(3):656–665, 2016a.

Huaping Liu, Yuanlong Yu, Fuchun Sun, and Jason Gu. Visual–tactile fusion for object recognition. *IEEE Transactions on Automation Science and Engineering*, 14(2):996–1008, 2016b.

Huaping Liu, Yupei Wu, Fuchun Sun, and Di Guo. Recent progress on tactile object recognition. *International Journal of Advanced Robotic Systems*, 14(4):1729881417717056, 2017.

Huaping Liu, Di Guo, Fuchun Sun, Wuqiang Yang, Steve Furber, and Tengchen Sun. Embodied tactile perception and learning. *Brain Science Advances*, 6(2):132–158, 2020.

Liu Liu, Han Xue, Wenqiang Xu, Haoyuan Fu, and Cewu Lu. Toward real-world category-level articulation pose estimation. *IEEE Transactions on Image Processing*, 31:1072–1083, 2022b.

Xingyu Liu, Gu Wang, Yi Li, and Xiangyang Ji. Catre: Iterative point clouds alignment for category-level object pose refinement. In *European Conference on Computer Vision (ECCV)*, 2022c.

John Lloyd and Nathan F Lepora. Goal-driven robotic pushing using tactile and proprioceptive feedback. *arXiv preprint arXiv:2012.01859*, 2020.

John Lloyd and Nathan F Lepora. Goal-driven robotic pushing using tactile and proprioceptive feedback. *IEEE Transactions on Robotics*, 38(2):1201–1212, 2021.

David G Lowe. Fitting parameterized three-dimensional models to images. *IEEE transactions on pattern analysis and machine intelligence*, 13(5):441–450, 1991.

Kendall Lowrey, Svetoslav Kolev, Jeremy Dao, Aravind Rajeswaran, and Emanuel Todorov. Reinforcement learning for non-prehensile manipulation: Transfer from simulation to physical system. In *2018 IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR)*, pages 35–42. IEEE, 2018.

Jingpei Lu, Florian Richter, and Michael C Yip. Markerless camera-to-robot pose estimation via self-supervised sim-to-real transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21296–21306, 2023.

Shan Luo, Wenxuan Mou, Kaspar Althoefer, and Hongbin Liu. Localizing the object contact through matching tactile features with visual map. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3903–3908. IEEE, 2015.

Shan Luo, Joao Bimbo, Ravinder Dahiya, and Hongbin Liu. Robotic tactile perception of object properties: A review. *Mechatronics*, 48:54–67, 2017.

Kevin M Lynch, Hitoshi Maekawa, and Kazuo Tanie. Manipulation and active sensing by pushing using tactile feedback. In *IROS*, volume 1, 1992.

Jeffrey Mahler, Florian T Pokorny, Brian Hou, Melrose Roderick, Michael Laskey, Mathieu Aubry, Kai Kohlhoff, Torsten Kröger, James Kuffner, and Ken Goldberg. Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 1957–1964. IEEE, 2016.

Martin Maier, Mahfuzulhoq Chowdhury, Bhaskar Prasad Rimal, and Dung Pham Van. The tactile internet: vision, recent progress, and open challenges. *IEEE Communications Magazine*, 54(5):138–145, 2016.

George Marsaglia. Choosing a point from the surface of a sphere. *The Annals of Mathematical Statistics*, 43(2):645–646, 1972.

Wolfram Martens, Yannick Poffet, Pablo Ramón Soria, Robert Fitch, and Salah Sukkarieh. Geometric priors for gaussian process implicit surfaces. *IEEE Robotics and Automation Letters*, 2(2):373–380, 2016.

Roberto Martín-Martín and Oliver Brock. Coupled recursive estimation for online interactive perception of articulated objects. *The International Journal of Robotics Research*, 41(8):741–777, 2022.

Gail Martino and Lawrence E Marks. Cross-modal interaction between vision and touch: the role of synesthetic correspondence. *Perception*, 29(6):745–754, 2000.

Matthew T Mason. Mechanics and planning of manipulator pushing operations. *The International Journal of Robotics Research*, 5(3):53–71, 1986a.

Matthew T Mason. On the scope of quasi-static pushing. In *Int. Symp. on Rob. Res., 1986*, pages 229–233, 1986b.

Paul M McDonnell. The development of visually guided reaching. *Perception & Psychophysics*, 18(3):181–185, 1975.

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

Martin Meier, Guillaume Walck, Robert Haschke, and Helge J Ritter. Distinguishing sliding from slipping during object pushing. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5579–5584. IEEE, 2016.

Azure Microsoft. Azure kinect ros driver, 2024. URL `https://github.com/microsoft/Azure_Kinect_ROS_Driver`.

Risto Miikkulainen, James A Bednar, Yoonsuck Choe, and Joseph Sirosh. *Computational maps in the visual cortex*. Springer Science & Business Media, 2006.

Chaitanya Mitash, Abdeslam Boularias, and Kostas Bekris. Physics-based scene-level reasoning for object pose estimation in clutter. *The International Journal of Robotics Research*, 41(6):615–636, 2022.

Rasoul Mojtahedzadeh, Abdelbaki Bouguerra, Erik Schaffernicht, and Achim J Lilienthal. Support relation analysis and decision making for safe robotic manipulation tasks. *Robotics and Autonomous Systems*, 71:99–117, 2015.

Mark Moll and Michael A Erdmann. Reconstructing shape from motion using tactile sensors. In *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems.*, volume 2, pages 692–700. IEEE, 2001.

Tomas Möller and Ben Trumbore. Fast, minimum storage ray-triangle intersection. *Journal of graphics tools*, 2(1):21–28, 1997.

Antonio Morales, Mario Prats, Pedro Sanz, and Angel P Pobil. An experiment in the use of manipulation primitives and tactile perception for reactive grasping. In *Robotics: Science and Systems, Workshop on Robot Manipulation: Sensing and Adapting to the Real World, Atlanta, USA*, 2007.

Douglas Morrison, Peter Corke, and Jürgen Leitner. Learning robust, real-time, reactive robotic grasping. *The International journal of robotics research*, 39(2-3):183–201, 2020.

Stephan Mühlbacher-Karrer, Juliana Padilha Leitzke, Lisa-Marie Faller, and Hubert Zangl. Non-iterative object detection methods in electrical tomography for robotic applications. *COMPEL-The international journal for computation and mathematics in electrical and electronic engineering*, 2017.

Prajval Kumar Murali. Reconstruction dataset, 2025. URL `https://www.robotac.eu/tactile-reconstruction`.

Prajval Kumar Murali, Michael Gentner, and Mohsen Kaboli. Active visuo-tactile point cloud registration for accurate pose estimation of objects in an unknown workspace. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2838–2844. IEEE, 2021.

Prajval Kumar Murali, Ravinder Dahiya, and Mohsen Kaboli. An empirical evaluation of various information gain criteria for active tactile action selection for pose estimation. In *2022 IEEE International Conference on Flexible and Printable Sensors and Systems (FLEPS)*, pages 1–4. IEEE, 2022a.

Prajval Kumar Murali, Anirvan Dutta, Michael Gentner, Etienne Burdet, Ravinder Dahiya, and Mohsen Kaboli. Active visuo-tactile interactive robotic perception for accurate object pose estimation in dense clutter. *IEEE Robotics and Automation Letters*, 2022b.

Prajval Kumar Murali, Mohsen Kaboli, and Ravinder Dahiya. Intelligent in-vehicle interaction technologies. *Advanced Intelligent Systems*, 4(2):2100122, 2022c.

Prajval Kumar Murali, Cong Wang, Ravinder Dahiya, and Mohsen Kaboli. Towards robust 3d object recognition with dense-to-sparse deep domain adaptation. In *2022 IEEE International Conference on Flexible and Printable Sensors and Systems (FLEPS)*, pages 1–4. IEEE, 2022d.

Prajval Kumar Murali, Cong Wang, Dongheui Lee, Ravinder Dahiya, and Mohsen Kaboli. Deep active cross-modal visuo-tactile transfer learning for robotic object recognition. *under Review IEEE Robotics and Automation Letters*, 2022e.

Prajval Kumar Murali, Bernd Porr, and Mohsen Kaboli. Touch if it's transparent! actor: Active tactile-based category-level transparent object reconstruction. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10792–10799. IEEE, 2023.

Prajval Kumar Murali, Bernd Porr, and Mohsen Kaboli. Shared visuo-tactile interactive perception for robust object pose estimation. *The International Journal of Robotics Research*, page 02783649241301443, 2024.

Richard M Murray, Zexiang Li, and S Shankar Sastry. *A mathematical introduction to robotic manipulation*. CRC press, 2017.

Yashraj Narang, Balakumar Sundaralingam, Miles Macklin, Arsalan Mousavian, and Dieter Fox. Sim-to-real for robotic tactile sensing via physics-based simulation and learned latent projections. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6444–6451. IEEE, 2021.

Nicolás Navarro-Guerrero, Sibel Toprak, Josip Josifovski, and Lorenzo Jamone. Visuo-haptic object perception for robots: an overview. *Autonomous Robots*, 47(4):377–403, 2023.

Howard R Nicholls and Mark H Lee. A survey of robot tactile sensing technology. *The International Journal of Robotics Research*, 8(3):3–30, 1989.

Kevin Nickels and Seth Hutchinson. Model-based tracking of complex articulated objects. *IEEE Transactions on Robotics and Automation*, 17(1):28–36, 2001.

OpenCV. Camera intrinsic calibration, 2024. URL `https://docs.opencv.org/4.x/dc/dbb/tutorial_py_calibration.html`.

Shaul Oron, Tali Dekel, Tianfan Xue, William T Freeman, and Shai Avidan. Best-buddies similarity—robust template matching using mutual nearest neighbors. *IEEE transactions on pattern analysis and machine intelligence*, 40(8):1799–1813, 2017.

G Dias Pais, Srikumar Ramalingam, Venu Madhav Govindu, Jacinto C Nascimento, Rama Chellappa, and Pedro Miraldo. 3dregnet: A deep neural network for 3d point registration.

In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7193–7203, 2020.

Antonio Paolillo, Kevin Chappellet, Anastasia Bolotnikova, and Abderrahmane Kheddar. Interlinked visual tracking and robotic manipulation of articulated objects. *IEEE Robotics and Automation Letters*, 3(4):2746–2753, 2018.

Yifei Pei, Lei Yan, Zuheng Wu, Jikai Lu, Jianhui Zhao, Jingsheng Chen, Qi Liu, and Xiaobing Yan. Artificial visual perception nervous system based on low-dimensional material photoelectric memristors. *ACS nano*, 15(11):17319–17326, 2021.

Stefano Pellegrini, Konrad Schindler, and Daniele Nardi. A generalisation of the icp algorithm for articulated bodies. In *BMVC*, volume 3, page 4, 2008.

Anna Petrovskaya and Oussama Khatib. Global localization of objects via touch. *IEEE Transactions on Robotics*, 27(3):569–585, 2011.

Zachary Pezzementi, Erion Plaku, Caitlin Reyda, and Gregory D Hager. Tactile-object recognition from appearance information. *IEEE Transactions on robotics*, 27(3):473–487, 2011.

Martin Pfanne, Maxime Chalon, Freek Stulp, and Alin Albu-Schäffer. Fusing joint measurements and visual features for in-hand object pose estimation. *IEEE Robotics and Automation Letters*, 3(4):3497–3504, 2018.

Cristina Piazza, Giorgio Grioli, Manuel G Catalano, and Antonio Bicchi. A century of robotic hands. *Annual Review of Control, Robotics, and Autonomous Systems*, 2(1): 1–32, 2019.

Lerrel Pinto and Abhinav Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 3406–3413. IEEE, 2016.

François Pomerleau, Francis Colas, Roland Siegwart, and Stéphane Magnenat. Comparing icp variants on real-world data sets. *Autonomous Robots*, 34(3):133–148, 2013.

Christian Potthast and Gaurav S Sukhatme. A probabilistic framework for next best view estimation in a cluttered environment. *Journal of Visual Communication and Image Representation*, 25(1):148–164, 2014.

Tony J Prescott, Mathew E Diamond, and Alan M Wing. Active touch sensing, 2011.

Matthew Purri and Kristin Dana. Teaching cameras to feel: Estimating tactile physical properties of surfaces from images. In *European Conference on Computer Vision*, pages 1–20. Springer, 2020.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

Haozhi Qi, Brent Yi, Sudharshan Suresh, Mike Lambeta, Yi Ma, Roberto Calandra, and Jitendra Malik. General in-hand object rotation with vision and touch. In *Conference on Robot Learning*, pages 2549–2564. PMLR, 2023.

Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan, 2009.

Carl Edward Rasmussen and Hannes Nickisch. Gaussian processes for machine learning (gpml) toolbox. *The Journal of Machine Learning Research*, 11:3011–3015, 2010.

Jan Razlaw, David Droeschel, Dirk Holz, and Sven Behnke. Evaluation of registration methods for sparse 3d laser scans. In *2015 european conference on mobile robots (ecmr)*, pages 1–7. IEEE, 2015.

Philippe Rochat, Elliott M Blass, and Lisa B Hoffmeyer. Oropharyngeal control of hand-mouth coordination in newborn infants. *Developmental psychology*, 24(4):459, 1988.

Alberto Rodriguez, Matthew T Mason, and Steve Ferry. From caging to grasping. *The International Journal of Robotics Research*, 31(7):886–900, 2012.

Lukas Rustler, Jens Lundell, Jan Kristof Behrens, Ville Kyrki, and Matej Hoffmann. Active visuo-haptic object shape completion. *IEEE Robotics and Automation Letters*, 7(2): 5254–5261, 2022.

Radu Bogdan Rusu and Steve Cousins. 3d is here: Point cloud library (pcl). In *2011 IEEE international conference on robotics and automation*, pages 1–4. IEEE, 2011.

Radu Bogdan Rusu, Zoltan Csaba Marton, Nico Blodow, Mihai Emanuel Dolha, and Michael Beetz. Functional object mapping of kitchen environments. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3525–3532. IEEE, 2008.

Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE international conference on robotics and automation*, pages 3212–3217, 2009.

Coralie Sann and Arlette Streri. Perception of object shape and texture in human newborns: evidence from cross-modal transfer tasks. *Developmental science*, 10(3):399–410, 2007.

Brad Saund, Shiyuan Chen, and Reid Simmons. Touch based localization of parts for high precision manufacturing. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 378–385. IEEE, 2017.

Henry Schaub and Alfred Schöttl. 6-dof grasp detection for unknown objects. In *2020 10th International Conference on Advanced Computer Information Technologies (ACIT)*, pages 400–403. IEEE, 2020.

Philipp Schmidt, Nikolaus Vahrenkamp, Mirko Wächter, and Tamim Asfour. Grasping of unknown objects using deep convolutional neural networks based on depth images. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 6831–6838. IEEE, 2018.

Tanner Schmidt, Richard Newcombe, and Dieter Fox. Dart: dense articulated real-time tracking with consumer depth cameras. *Autonomous Robots*, 39:239–258, 2015.

Alexander Schmitz, Perla Maiolino, Marco Maggiali, Lorenzo Natale, Giorgio Cannata, and Giorgio Metta. Methods and technologies for the implementation of large-scale robot tactile sensors. *IEEE Transactions on Robotics*, 27(3):389–400, 2011.

Max Schwarz, Christian Lenz, Germán Martín García, Seongyong Koo, Arul Selvam Periyasamy, Michael Schreiber, and Sven Behnke. Fast object learning and dual-arm coordination for cluttered stowing, picking, and packing. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3347–3354. IEEE, 2018.

Lucia Seminara, Marco Capurro, Paolo Cirillo, Giorgio Cannata, and Maurizio Valle. Electromechanical characterization of piezoelectric pvdf polymer films for tactile sensors in robotics applications. *Sensors and Actuators A: Physical*, 169(1):49–58, 2011.

Claude Elwood Shannon. A mathematical theory of communication. *ACM SIGMOBILE mobile computing and communications review*, 5(1):3–55, 2001.

David Silvera Tawil, David Rye, and Mari Velonaki. Interpretation of the modality of touch on an artificial arm covered with an eit-based sensitive skin. *The International Journal of Robotics Research*, 31(13):1627–1641, 2012.

David Silvera-Tawil, David Rye, Manuchehr Soleimani, and Mari Velonaki. Electrical impedance tomography for artificial sensitive robotic skin: A review. *IEEE Sensors Journal*, 15(4):2001–2016, 2014.

Edward Smith, Roberto Calandra, Adriana Romero, Georgia Gkioxari, David Meger, Jitendra Malik, and Michal Drozdzal. 3d shape reconstruction from vision and touch. *Advances in Neural Information Processing Systems*, 33:14193–14206, 2020.

Joan Sola, Jeremie Deray, and Dinesh Atchuthan. A micro lie theory for state estimation in robotics. *arXiv preprint arXiv:1812.01537*, 2018.

Jae S Son, R Howe, Jonathan Wang, and Gregory D Hager. Preliminary results on grasping with vision and touch. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS'96*, volume 3, pages 1068–1075. IEEE, 1996.

Arlette Streri. Tactile discrimination of shape and intermodal transfer in 2-to 3-month-old infants. *British Journal of Developmental Psychology*, 5(3):213–220, 1987.

Arlette Streri and Edouard Gentaz. Cross-modal recognition of shape from hand to eyes and handedness in human newborns. *Neuropsychologia*, 42(10):1365–1369, 2004.

Arlette Streri and Sylvie Milhet. Equivalences intermodales de la forme des objets entre la vision et le toucher chez les bébés de 2 mois. *L'Année psychologique*, 88(3):329–341, 1988.

Arlette Streri and Marie-Germaine Pêcheux. Tactual habituation and discrimination of form in infancy: A comparison with vision. *Child Development*, pages 100–104, 1986.

Tricia Striano and Emily W Bushnell. Haptic perception of material properties by 3-month-old infants. *Infant Behavior and Development*, 28(3):266–289, 2005.

Claudius Strub, Florentin Wörgötter, Helge Ritter, and Yulia Sandamirskaya. Correcting pose estimates during tactile exploration of object shape: a neuro-robotic study. In *4th International Conference on Development and Learning and on Epigenetic Robotics*, pages 26–33. IEEE, 2014.

Jochen Stüber, Claudio Zito, and Rustam Stolkin. Let's push things forward: A survey on robot pushing. *Frontiers in Robotics and AI*, 7:8, 2020.

Jürgen Sturm, Cyrill Stachniss, and Wolfram Burgard. A probabilistic framework for learning kinematic models of articulated objects. *Journal of Artificial Intelligence Research*, 41:477–526, 2011.

Zhiqiang Sui, Lingzhu Xiang, Odest C Jenkins, and Karthik Desingh. Goal-directed robot manipulation through axiomatic scene estimation. *The International Journal of Robotics Research*, 36(1):86–104, 2017.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450. Springer, 2016.

Baochen Sun, Jiashi Feng, and Kate Saenko. Return of frustratingly easy domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

Yu Sun and John M Hollerbach. Active robot calibration algorithm. In *2008 IEEE international conference on robotics and automation*, pages 1276–1281. IEEE, 2008.

Sudharshan Suresh, Maria Bauza, Kuan-Ting Yu, Joshua G Mangelson, Alberto Rodriguez, and Michael Kaess. Tactile slam: Real-time inference of shape and pose from planar pushing. *arXiv preprint arXiv:2011.07044*, 2020.

Sudharshan Suresh, Maria Bauza, Kuan-Ting Yu, Joshua G Mangelson, Alberto Rodriguez, and Michael Kaess. Tactile slam: Real-time inference of shape and pose from planar pushing. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11322–11328. IEEE, 2021.

Kuniyuki Takahashi and Jethro Tan. Deep visuo-tactile learning: Estimation of tactile properties from images. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8951–8957. IEEE, 2019.

William Taube Navaraj, Carlos García Núñez, Dhayalan Shakthivel, Vincenzo Vinciguerra, Fabrice Labeau, Duncan H Gregory, and Ravinder Dahiya. Nanowire fet based neural element for robotic tactile sensing skin. *Frontiers in neuroscience*, 11:501, 2017.

M Lamine Tazir, Tawsif Gokhool, Paul Checchin, Laurent Malaterre, and Laurent Trassoudaine. Cicp: Cluster iterative closest point for sparse–dense point cloud registration. *Robotics and Autonomous Systems*, 108:66–86, 2018.

Meng Tian, Marcelo H Ang, and Gim Hee Lee. Shape prior deformation for categorical 6d object pose and size estimation. In *European Conference on Computer Vision*, pages 530–546. Springer, 2020.

Tito Pradhono Tomo. *Development of a Compact, Soft, Distributed 3-axis Hall Effect-based Skin Sensor: uSkin*. PhD thesis, Waseda University, 2019.

Tito Pradhono Tomo, Alexander Schmitz, Wai Keat Wong, Harris Kristanto, Sophon Somlor, Jinsun Hwang, Lorenzo Jamone, and Shigeki Sugano. Covering a robot fingertip with uskin: A soft electronic skin with distributed 3-axis force sensitive elements for robot hands. *IEEE Robotics and Automation Letters*, 3(1):124–131, 2017.

Tito Pradhono Tomo, Massimo Regoli, Alexander Schmitz, Lorenzo Natale, Harris Kristanto, Sophon Somlor, Lorenzo Jamone, Giorgio Metta, and Shigeki Sugano. A new silicone structure for uskin—a soft, distributed, digital 3-axis skin sensor and its integration on the humanoid robot icub. *IEEE Robotics and Automation Letters*, 3(3):2584–2591, 2018.

Sibel Toprak, Nicolás Navarro-Guerrero, and Stefan Wermter. Evaluating integration strategies for visuo-haptic object recognition. *Cognitive computation*, 10:408–425, 2018.

Niccoló Tosi, Olivier David, and Herman Bruyninckx. Action selection for touch-based localisation trading off information gain and execution time. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2270–2275. IEEE, 2014.

Roger Y Tsai and Reimar K Lenz. A new technique for fully autonomous and efficient 3 d robotics hand/eye calibration. *IEEE Transactions on robotics and automation*, 5(3): 345–358, 1989.

Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 13(04): 376–380, 1991.

John Van der Kamp, Raôul Oudejans, and Geert Savelsbergh. The development and learning of the visual control of movement: An ecological perspective. *Infant Behavior and Development*, 26(4):495–515, 2003.

Laurens Van Der Maaten. Accelerating t-sne using tree-based algorithms. *The journal of machine learning research*, 15(1):3221–3245, 2014.

Rudolph Van Der Merwe, Arnaud Doucet, Nando De Freitas, and Eric A Wan. The unscented particle filter. In *Advances in neural information processing systems*, pages 584–590, 2001.

G. Vezzani, N. Jamali, U. Pattacini, G. Battistelli, L. Chisci, and L. Natale. A novel bayesian filtering approach to tactile object recognition. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 256–263, 2016. doi: 10.1109/HUMANOIDS.2016.7803286.

Giulia Vezzani, Ugo Pattacini, Giorgio Battistelli, Luigi Chisci, and Lorenzo Natale. Memory unscented particle filter for 6-dof tactile localization. *IEEE Transactions on Robotics*, 33(5):1139–1155, 2017.

Joshua Wade, Tapomayukh Bhattacharjee, Ryan D Williams, and Charles C Kemp. A force and thermal sensing skin for robots in human environments. *Robotics and Autonomous Systems*, 96:1–14, 2017.

Kenneth J Waldron and James Schmiedeler. Kinematics. *Springer handbook of robotics*, pages 11–36, 2016.

R Walkler. Developments in dextrous hands for advanced robotic applications. In *Proc. the Sixth Biannual World Automation Congress, Seville, Spain*, pages 123–128, 2004.

Changjin Wan, Pingqiang Cai, Xintong Guo, Ming Wang, Naoji Matsuhisa, Le Yang, Zhisheng Lv, Yifei Luo, Xian Jun Loh, and Xiaodong Chen. An artificial sensory neuron with visual-haptic fusion. *Nature communications*, 11(1):4602, 2020.

He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2642–2651, 2019.

Pengyuan Wang, HyunJun Jung, Yitong Li, Siyuan Shen, Rahul Parthasarathy Srikanth, Lorenzo Garattoni, Sven Meier, Nassir Navab, and Benjamin Busam. Phocal: A multimodal dataset for category-level object pose estimation with photometrically challenging objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21222–21231, 2022a.

Shaoxiong Wang, Jiajun Wu, Xingyuan Sun, Wenzhen Yuan, William T Freeman, Joshua B Tenenbaum, and Edward H Adelson. 3d shape perception from monocular vision, touch, and shape priors. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1606–1613. IEEE, 2018.

Shaoxiong Wang, Mike Lambeta, Po-Wei Chou, and Roberto Calandra. Tacto: A fast, flexible, and open-source simulator for high-resolution vision-based tactile sensors. *IEEE Robotics and Automation Letters*, 7(2):3930–3937, 2022b.

Weiss. Weiss robotics, 2025. URL `https://weiss-robotics.com/company/`.

Greg Welch and Gary Bishop. An introduction to the kalman filter. 1995.

Bowen Wen and Kostas Bekris. Bundletrack: 6d pose tracking for novel objects without instance or category-level 3d models. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8067–8074. IEEE, 2021.

Andrew T Woods and Fiona N Newell. Visual, haptic and cross-modal recognition of objects and scenes. *Journal of physiology-Paris*, 98(1-3):147–159, 2004.

Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, Li Yi, Angel X. Chang, Leonidas J. Guibas, and Hao Su. SAPIEN: A simulated part-based interactive environment. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Haoyu Xiong, Russell Mendonca, Kenneth Shaw, and Deepak Pathak. Adaptive mobile manipulation for articulated objects in the open world. *arXiv preprint arXiv:2401.14403*, 2024.

Zhenjia Xu, Zhanpeng He, and Shuran Song. Universal manipulation policy network for articulated objects. *IEEE robotics and automation letters*, 7(2):2447–2454, 2022.

Akihiko Yamaguchi and Christopher G Atkeson. Optical skin for robots: Tactile sensing and whole-body vision. In *Workshop on Tactile Sensing for Manipulation, Robotics: Science and Systems (RSS)*, volume 25, pages 133–134, 2017.

Ke Yan. Domain-adaptation-toolbox: Wrappers and implementations of several domain adaptation / transfer learning / semi-supervised learning algorithms, 2024. URL `https://github.com/viggin/domain-adaptation-toolbox`.

Bin Yang, Patrick Pfreundschuh, Roland Siegwart, Marco Hutter, Peyman Moghadam, and Vaishakh Patil. Tulip: Transformer for upsampling of lidar point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15354–15364, 2024.

Guang-Zhong Yang, Jim Bellingham, Pierre E Dupont, Peer Fischer, Luciano Floridi, Robert Full, Neil Jacobstein, Vijay Kumar, Marcia McNutt, and Robert Merrifield. The grand challenges of science robotics. *Science robotics*, 3(14):eaar7650, 2018.

Heng Yang, Jingnan Shi, and Luca Carlone. Teaser: Fast and certifiable point cloud registration. *IEEE Transactions on Robotics*, 37(2):314–333, 2020.

Zhenpei Yang, Jeffrey Z Pan, Linjie Luo, Xiaowei Zhou, Kristen Grauman, and Qixing Huang. Extreme relative pose estimation for rgb-d scans via scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4531–4540, 2019.

A Yao and Manuchehr Soleimani. A pressure mapping imaging device based on electrical impedance tomography of conductive fabrics. *Sensor Review*, 2012.

Zhengkun Yi, Roberto Calandra, Filipe Veiga, Herke van Hoof, Tucker Hermans, Yilei Zhang, and Jan Peters. Active tactile object exploration with gaussian processes. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4925–4930. IEEE, 2016.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems*, 27: 3320–3328, 2014.

Kuan-Ting Yu, John Leonard, and Alberto Rodriguez. Shape and pose recovery from planar pushing. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1208–1215. IEEE, 2015.

Kuan-Ting Yu, Maria Bauza, Nima Fazeli, and Alberto Rodriguez. More than a million ways to be pushed. a high-fidelity experimental dataset of planar pushing. In *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 30–37. IEEE, 2016.

Lequan Yu, Xianzhi Li, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. Pu-net: Point cloud upsampling network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2790–2799, 2018.

Wentao Yuan, Tejas Khot, David Held, Christoph Mertz, and Martial Hebert. Pcn: Point completion network. In *2018 international conference on 3D vision (3DV)*, pages 728–737. IEEE, 2018.

Wenzhen Yuan, Rui Li, Mandayam A Srinivasan, and Edward H Adelson. Measurement of shear and slip with a gelsight tactile sensor. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 304–311. IEEE, 2015.

Pietro Zanuttigh, Giulio Marin, Carlo Dal Mutto, Fabio Dominio, Ludovico Minto, and Guido Maria Cortelazzo. Time-of-flight and structured light depth cameras. *Technology and Applications*, 978(3), 2016.

Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1802–1811, 2017.

Andy Zeng, Shuran Song, Stefan Welker, Johnny Lee, Alberto Rodriguez, and Thomas Funkhouser. Learning synergies between pushing and grasping with self-supervised deep reinforcement learning. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4238–4245. IEEE, 2018.

Fuzhen Zhang. Quaternions and matrices of quaternions. *Linear algebra and its applications*, 251:21–57, 1997.

Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *International conference on machine learning*, pages 7354–7363. PMLR, 2019.

Hanbo Zhang, Yunfan Lu, Cunjun Yu, David Hsu, Xuguang La, and Nanning Zheng. Invigorate: Interactive visual grounding and grasping in clutter. In *2021 Robotics Science and Systems Conference (RSS)*, 2021.

Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Ting Zhang, Hong Liu, Li Jiang, Shaowei Fan, and Jing Yang. Development of a flexible 3-d tactile sensor system for anthropomorphic artificial hand. *IEEE sensors Journal*, 13 (2):510–518, 2012.

Wendong Zheng, Huaping Liu, Bowen Wang, and Fuchun Sun. Cross-modal material perception for novel objects: a deep adversarial learning method. *IEEE Transactions on Automation Science and Engineering*, 17(2):697–707, 2019.

Jiaji Zhou, Matthew T Mason, Robert Paolini, and Drew Bagnell. A convex polynomial model for planar sliding mechanics: theory, application, and experimental validation. *The International Journal of Robotics Research*, 37(2-3):249–265, 2018a.

Linglong Zhou, Guoxin Wu, Yunbo Zuo, Xuanyu Chen, and Hongle Hu. A comprehensive review of vision-based 3d reconstruction methods. *Sensors*, 24(7):2314, 2024.

Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A modern library for 3D data processing. *arXiv:1801.09847*, 2018b.

Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3d: A modern library for 3d data processing. *arXiv preprint arXiv:1801.09847*, 2018c.

Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3):257–276, 2023.

Appendix A

Mathematical Background

This chapter recalls fundamental mathematical results from the literature that are used throughout the thesis.

A.1 Homogeneous Transformations

A coordinate frame is described by its origin point and orientation and described in this thesis using a capitalised letter such as A. Frames can move in time with respect to a reference frame and can be used to describe the pose, which is the position and orientation, of a rigid body with respect to the reference frame. Given two frames A and B, we employ the notation ${}^{A}R_{B} \in SO(3)$ to denote the rotation of frame B to frame A. Similarly, we denote ${}^{A}t_{B} \in \mathbb{R}^3$ as the translation of the point of origin of frame B to the origin of frame A. To describe the pose of frame B with respect to frame A, a homogeneous transformation matrix is defined as follows:

$$ {}^{A}H_{B} = \begin{bmatrix} {}^{A}R_{B} & {}^{A}t_{B} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \in SE(3) \tag{A.1} $$

The homogeneous transformation matrix can also be applied to a point in 3D space. Let a point be described with respect to frame A be denoted as ${}^{A}p$. Let ${}^{A}\bar{p}$ be the homogeneous representation of the point ${}^{A}p$ i.e., ${}^{A}\bar{p} = ({}^{A}p; 1) \in \mathbb{R}^4$. Then the following relation holds true:

$$ {}^{A}\bar{p} = {}^{A}H_{B}\,{}^{B}\bar{p} \tag{A.2} $$

Equivalently, the Eq. (A.2) can be expressed as ${}^{A}p = {}^{A}R_{B}\,{}^{B}p + {}^{A}t_{B}$. The inverse of a homogeneous transform is given by:

$$ {}^{B}H_{A} = {}^{A}H_{B}^{-1} = \begin{bmatrix} {}^{A}R_{B}^{T} & -{}^{A}R_{B}^{T}\,{}^{A}t_{B} \\ 0_{1\times 3} & 1 \end{bmatrix} \tag{A.3} $$

Homogeneous matrices can be composed together when performing one transformation after another as follows:

$$^{A}H_{C} = {}^{A}H_{B}{}^{B}H_{C} \tag{A.4}$$

Note that all multiplications between matrices denote standard matrix multiplication unless specified otherwise. The set of all matrices described in Eq. A.1 span the so called SE(3), Special Euclidean Lie Group of rigid body displacements in 3D. There are sub-groups of SE(3) of importance such as SO(3) special orthogonal group of 3 dimensions, which is a group of all possible 3D rotations. The interested reader is referred to Waldron and Schmiedeler (2016) for further details on transformation matrices and their role in robotics and computer vision. Lie groups and algebra are introduced in Sec. A.2.

A.1.1 Quaternions

Rotations in SO(3) can be parameterized in multiple different ways, such as Euler angles, rotation matrices, unit quaternions, angle-axis representations, and so on. Rotation matrices and, in particular, quaternions are used in this thesis which offer robustness against issues such as the gimbal lock due to Euler angle parameterisation (Zhang, 1997). Some important properties of quaternions are detailed in this section.

A quaternion, $\mathbf{q}$ is represented as:

$$\mathbf{q} = \overbrace{w}^{\text{real}} + \overbrace{x\mathbf{i} + y\mathbf{j} + z\mathbf{k}}^{\text{vector}} \tag{A.5}$$

where $w, x, y, z \in \mathbb{R}$ and $\mathbf{i}, \mathbf{j}, \mathbf{k}$ are the basis vectors.

Two quaternions are added by adding the respective components separately as:

$$\mathbf{q_1} + \mathbf{q_2} = (w_1 + w_2) + (x_1 + x_2)\mathbf{i} + (y_1 + y_2)\mathbf{j} + (z_1 + z_2)\mathbf{k} \tag{A.6}$$

Quaternion summation is associative, commutative, and distributive.

The multiplication of quaternions is denoted by $\odot$ and is given as:

$$\begin{aligned} \mathbf{q_1} \odot \mathbf{q_2} = &(w_1 w_2 - x_1 x_2 - y_1 y_2 - z_1 z_2) \\ &+ (w_1 x_2 + x_1 w_2 + y_1 z_2 - z_1 y_2)\mathbf{i} \\ &+ (w_1 y_2 + y_1 w_2 + z_1 x_2 - x_1 z_2)\mathbf{j} \\ &+ (w_1 z_2 + z_1 w_2 + x_1 y_2 - y_1 x_2)\mathbf{k} \end{aligned} \tag{A.7}$$

Equivalently, quaternion multiplication is also provided in matrix form as:

$$\mathbf{q}_1 \odot \mathbf{q}_2 = \begin{bmatrix} w_1 & -\vec{\mathbf{q}_1}^T \\ \vec{\mathbf{q}_1} & \vec{\mathbf{q}_1}^{\times} + w_1 \mathbb{I}_3 \end{bmatrix} \mathbf{q}_2 \tag{A.8}$$

$$= \begin{bmatrix} w_2 & -\vec{\mathbf{q}_2}^T \\ \vec{\mathbf{q}_2} & -\vec{\mathbf{q}_2}^{\times} + w_2 \mathbb{I}_3 \end{bmatrix} \mathbf{q}_1 \tag{A.9}$$

where $\vec{q}$ is the vector part of the quaternion and $[v]^{\times}$ is the skew-symmetric matrix formed from the vector v.

The conjugate of a quaternion $\mathbf{q}^*$ is given by:

$$\mathbf{q}^* = w - x\mathbf{i} - y\mathbf{j} - z\mathbf{k} \tag{A.10}$$

The norm of a quaternion is given by:

$$||\mathbf{q}|| = \sqrt{\mathbf{q} \odot \mathbf{q}^*} = \sqrt{(w^2 + x^2 + y^2 + z^2)} \tag{A.11}$$

A unit quaternion denoted by $||\mathbf{q}|| = 1$ represents any rotation in SO(3) where the positive and negative basis vectors represent the same rotation.

For any unit quaternion $\mathbf{q}$, the operation $\mathbf{qpq}^*$ performs the rotation of a vector $\mathbf{p}$ about the direction $(x, y, z)^T$.

A.2 Lie Group and Algebra

A Lie group is a smooth differentiable manifold whose elements adhere to the axioms inherent to group theory. A differentiable or smooth manifold is a topological space that locally resembles linear space. The smoothness of the manifold implies the existence of a unique tangent space at each point.

A group $(G, \circ)$ is a set, G, with a composition operation $\circ$ that, for elements $X, Y, Z \in G$, satisfies the following axioms (Sola et al., 2018):

- Closure under $\circ$: $X \circ Y \in G$
- Identity $\mathcal{E}$: $\mathcal{E} \circ X = X \circ \mathcal{E} = X$
- Inverse X^{-1} : $X^{-1} \circ X = X \circ X^{-1} = \mathcal{E}$
- Associativity : $(X \circ Y) \circ Z = X \circ (Y \circ Z)$

Given $X(t)$ a point moving on a Lie group's manifold $\mathcal{M}$, its velocity $\dot{X} = \partial X / \partial t$ belongs to the space tangent to $\mathcal{M}$ at X, which is denoted as $T_X\mathcal{M}$. The smoothness of the

manifold implies the existence of a unique tangent space at each point. The structure of such tangent spaces is the same everywhere (Sola et al., 2018).

The tangent space at the identity, $T_{\mathcal{E}}\mathcal{M}$, is called the Lie algebra of $\mathcal{M}$ and is denoted as $\mathfrak{m}$.

There are two functions associated with the Lie group and its algebra:

- Exponential map: $exp : \mathfrak{m} \to \mathcal{M}$ converts elements of the Lie algebra into elements of the group (also denoted by $\varphi(\cdot)$).
- Logarithmic map: $log : \mathcal{M} \to \mathfrak{m}$ converts elements of the Lie group to the elements of the Lie algebra (also denoted by $\varphi^{-1}(\cdot)$).

A detailed introduction to Lie theory in the context of robotics is available in (Sola et al., 2018).

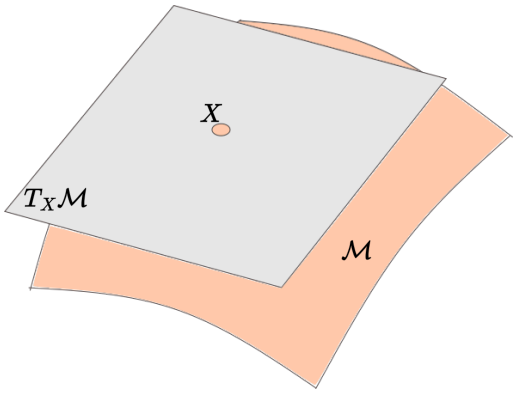


Figure A.1: A manifold $\mathcal{M}$ and the vector space $T_X\mathcal{M}$ tangent at the point X

A.3 Bayesian Filter

Filtering is a process of estimating the value of a quantity at time t using measurements up to and including time t. A recursive Bayesian Filter is a probabilistic approach to estimate the *a posteriori* probability of a state variable $\mathbf{x}$ recursively over time t using measurements $\mathbf{z}$ and possible control inputs $\mathbf{u}$. Two important assumptions are used in Bayesian Filtering: (a) the state follows a first-order Markov process i.e., $p(\mathbf{x}_t|\mathbf{x}_{0:t-1}) = p(\mathbf{x}_t|\mathbf{x}_{t-1})$ and (b) the measurements $\mathbf{z}_t$ are independent of all prior states apart from the current state $\mathbf{x_t}$.

From Bayes rule, we get the following relation:

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) = \frac{p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{z}_{1:t-1})}{p(\mathbf{z}_t|\mathbf{z}_{1:t-1})} \propto \eta\, p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{z}_{1:t-1}) \tag{A.12}$$

as the denominator $p(\mathbf{z}_t|\mathbf{z}_{1:t-1}) = \int p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{z}_{1:t-1})d\mathbf{x}_t$ is constant with respect to $\mathbf{x}$ and can be replaced by a normalisation constant η in practice. If the state and measurement variables are Gaussian distributed and the transition models (process and measurement model which define the state transition and the mapping of the state into the measurement space, respectively) are linear, then the recursive Bayesian Filter simplifies into a Kalman filter.

A.3.1 Kalman Filter

The Kalman Filter is a type of Bayesian Filter that aims to estimate the state $\mathbf{x} \in \mathbb{R}^n$ of a discrete-time process given as:

$$\mathbf{x}_{t+1} = \mathbf{A}_t\mathbf{x}_t + \mathbf{B}_t\mathbf{u}_t + \mathbf{w}_t \tag{A.13}$$

with measurements $\mathbf{z} \in \mathbb{R}^m$ that is governed by:

$$\mathbf{z}_t = \mathbf{H}_t\mathbf{x}_t + \mathbf{v}_t \tag{A.14}$$

The variables $\mathbf{u}_t$ represents the control input, $\mathbf{w}_t$ and $\mathbf{v}_t$ represents the process and measurement noise, the matrices $\mathbf{A}_t$ represents the process model, $\mathbf{B}_t$ relates the control input $\mathbf{u}_t$ to the state vector $\mathbf{x}_t$ and $\mathbf{H}_t$ represents the measurement model. The process and measurement noise is assumed to be zero-mean gaussian noise as:

$$\begin{aligned} p(\mathbf{w}) &\sim \mathcal{N}(0, \mathbf{Q}) \\ p(\mathbf{v}) &\sim \mathcal{N}(0, \mathbf{R}) \end{aligned} \tag{A.15}$$

The Kalman Filter estimates the state variable at time step t and then corrects the estimate from the noisy measurements. Hence, there are two discrete steps: prediction step (process update or time update) and update step (measurement update). The prediction step projects forward (in time) the current state and error covariance estimates to obtain the *a priori* estimates for the next time step. The Kalman Filter is a recursive filter, in each iteration a prediction and update step is performed and the *a posteriori* estimate is used for the next iteration. Note that the *a priori* estimate is denoted with the subscripts $()_{t+1|t}$ whereas the *a posteriori* estimate is denoted by $()_{t+1|t+1}$. The Kalman Filter is initialised with an value for state and the associated uncertainty as $\mathbf{x}_0, \mathbf{P}_0$. Being a recursive filter, only the estimates of the previous state and the current measurements are required to estimate the current value of the state variable. The prediction step estimates the *a priori* state value by propagating through the process model. The prediction step is provided as:

$$\mathbf{x}_{t+1|t} = \mathbf{A}\mathbf{x}_{t|t} + \mathbf{B}\mathbf{u}_t \tag{A.16}$$

$$\mathbf{P}_{t+1|t} = \mathbf{A}\mathbf{P}_{t|t}\mathbf{A}^T + \mathbf{Q}_t \tag{A.17}$$

During the update step, the Kalman gain $\mathbf{K}_t$ is computed which is used to update the *a posteriori* state estimate ($\mathbf{x}_{t+1|t+1}$) with the measurements values. The Kalman gain is computed as follows:

$$\mathbf{K}_t = \mathbf{P}_{t+1|t}\mathbf{H}^T(\mathbf{H}_t\mathbf{P}_{t+1|t}\mathbf{H}_t^T + \mathbf{R}_t)^{-1} \tag{A.18}$$

The *a posteriori* state and uncertainty estimate are computed as follows:

$$\mathbf{x}_{t+1|t+1} = \mathbf{x}_{t+1|1} + \mathbf{K}_t(\mathbf{z}_t - \mathbf{H}_t\mathbf{x}_{t+1|t}) \tag{A.19}$$

$$\mathbf{P}_{t+1|t+1} = (\mathbf{I} - \mathbf{K}_t\mathbf{H}_t)\mathbf{P}_{t+1|t} \tag{A.20}$$

The term $\mathbf{z}_t - \mathbf{H}_t\mathbf{x}_{t+1|t}$ is called the innovation which defines the residual between the observed measurements and the predicted measurements based on *a priori* state information.

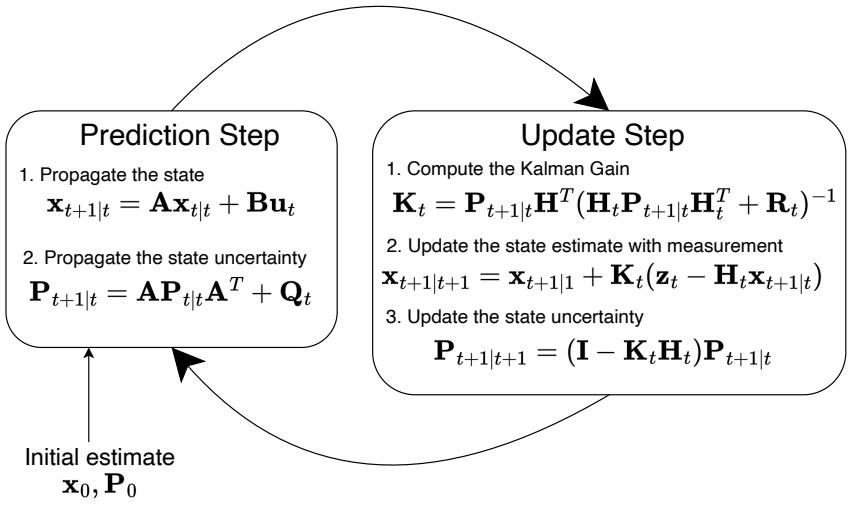


Figure A.2: Kalman filter operations

For tuning the Kalman Filter, the ratio between the process noise and the measurement noise determines whether the filter follows closer to the process model, if the process noise is smaller, or closer to the measurements, if the measurement noise is smaller. The Kalman Filter assumes linear process and measurement models. In case of non-linear models, an Extended Kalman Filter (EKF) or Unscented Kalman Filter (UKF) are generally used. The Kalman Filtering process and equations are shown in Fig. A.2. A detailed and rigorous

introduction to Kalman Filter is available in (Welch and Bishop, 1995).

A.3.2 Unscented Kalman Filter in Euclidean Space

Consider a system with known initial mean and initial covariance as $\bar{\mathbf{x}}_0 = E[\mathbf{x}_0]$ and $\mathbf{P}_0 = P[\mathbf{x}_0]$ respectively. The aim is to obtain the *a posteriori* state estimate $\hat{\mathbf{x}}_{k|k}$ and *a posteriori* covariance $\hat{\mathbf{P}}_{k|k}$ upto time k integrating k observations. The state $\hat{\mathbf{x}}_{k|k} \in SE(3)$ denotes the 6 DoF pose of the object. The state transitions model (also called process model) is denoted as $f(\cdot)$ and the measurement model as $h(\cdot)$. The *a posteriori* state can be obtained as follows:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}(\mathbf{z}_{k|k} - \hat{\mathbf{z}}_{k|k-1}) \tag{A.21}$$

wherein, $\mathbf{K}$ is the Kalman gain, $\hat{p}$ refers to prediction of variable p, and $\mathbf{z}$ is the measurement observation. When $f(\cdot)$ and $h(\cdot)$ are non-linear functions, an Unscented Kalman Filter (UKF) can be used. The UKF uses the *Unscented Transformation* technique which picks a minimal set of sample points, called sigma points around the mean. These sigma points are propagated through the non-linear functions from which a new mean and covariance is obtained. Consider the state as M dimensions, the sigma points and the corresponding weights around the mean of the state are calculated as follows:

$$\begin{aligned}
\sigma^0 &= \bar{\mathbf{x}} \\
\sigma^i &= \bar{\mathbf{x}} + (\sqrt{(M+\lambda)P})_i \quad i = 1,\dots M \\
\sigma^i &= \bar{\mathbf{x}} - (\sqrt{(M+\lambda)P})_{i-M} \quad i = M+1,\dots 2M \\
W_0^{(m)} &= \lambda/(M+\lambda) \\
W_0^{(c)} &= \lambda/(M+\lambda) + (1-\alpha^2+\beta) \\
W_i^{(m)} &= W_i^{(c)} = 1/2(M+\lambda) \quad i = 1,\dots 2M
\end{aligned} \tag{A.22}$$

where $\lambda = \{\alpha^2(M+\kappa) - M\}$, and α, β, κ are scaling parameters, $(\sqrt{(M+\lambda)P_i})$ refers to i^{th} column of the matrix square root. Propagating the sigma points through the state transition model, the predicted mean and covariance are obtained as:

$$\begin{aligned}
\hat{\mathbf{x}}_{k|k-1} &= \sum_{i=0}^{2M} W_i^{(m)} f(\sigma^i) \\
\mathbf{P}_{k|k-1} &= \sum_{i=0}^{2M} W_i^{(c)} (f(\sigma^i) - \hat{\mathbf{x}}_{k|k-1})(f(\sigma^i) - \hat{\mathbf{x}}_{k|k-1})^T + \mathbf{Q}_k
\end{aligned} \tag{A.23}$$

where $\mathbf{Q}_k$ is the covariance matrix of the state transition noise.

Similarly, the sigma points are transformed by the measurement model h as:

$$\begin{aligned}\hat{\mathbf{z}}_{k|k-1} &= \sum_{i=0}^{2M} W_i^{(m)} h(\sigma^i) \\ \mathbf{P}_{zz} &= \sum_{i=0}^{2M} W_i^{(c)} (h(\sigma^i) - \hat{\mathbf{z}}_{k|k-1})(h(\sigma^i) - \hat{\mathbf{z}}_{k|k-1})^T + \mathbf{R}_k\end{aligned} \tag{A.24}$$

where $\mathbf{R}_k$ is the covariance matrix of the observation noise.

In addition, the cross-covariance matrix is provided by:

$$\mathbf{P}_{xz} = \sum_{i=0}^{2M} W_i^{(c)} (f(\sigma)_i - \hat{\mathbf{x}}_{k|k-1})(h(\sigma)_i - \hat{\mathbf{z}}_{k|k-1})^T \tag{A.25}$$

The Kalman gain is computed as:

$$\mathbf{K}_k = \mathbf{P}_{xy}\mathbf{P}_{zz}^{-1} \tag{A.26}$$

Hence, the *a posteriori* mean and covariance matrix are obtained as:

$$\begin{aligned}\hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}(\mathbf{z}_{k|k} - \hat{\mathbf{z}}_{k|k-1}) \\ \mathbf{P}_{k|k} &= \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{P}_{yy} \mathbf{K}_k^T\end{aligned} \tag{A.27}$$

Appendix B

Derivation for Kullback–Leibler (KL) Divergence for Gaussian Distributions

The derivation for the Kullback–Leibler divergence for multivariate Gaussian distributions is provided below (Duchi, 2007).

Note that the KL divergence between two distributions P and Q is defined as

$$D_{KL}(P\|Q) = \mathrm{E}_P\left[\log\frac{P}{Q}\right],$$

where E_P refers to the expectation of P. The density function for a multivariate Gaussian (normal) distribution with mean μ and covariance matrix Σ is given by

$$p(x) = \frac{1}{(2\pi)^{n/2}\det(\Sigma)^{1/2}}\exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right).$$

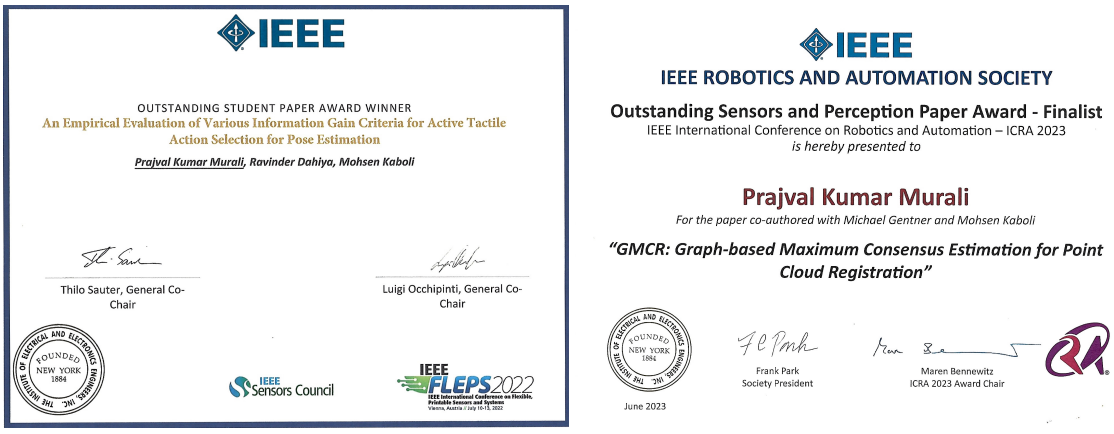
Now, considering two multivariate Gaussians in $\mathbb{R}^n$ namely, P_1 and P_2. We have

$$
\begin{aligned}
&D(P_1 \| P_2) \\
&= \mathrm{E}_{P_1}[\log P_1 - \log P_2] \\
&= \frac{1}{2}\mathrm{E}_{P_1}\left[-\log\det\Sigma_1 - (x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1) + \log\det\Sigma_2 + (x-\mu_2)^T\Sigma_2^{-1}(x-\mu_2)\right] \\
&= \frac{1}{2}\log\frac{\det\Sigma_2}{\det\Sigma_1} + \frac{1}{2}\mathrm{E}_{P_1}\left[-(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1) + (x-\mu_2)^T\Sigma_2^{-1}(x-\mu_2)\right] \\
&= \frac{1}{2}\log\frac{\det\Sigma_2}{\det\Sigma_1} + \frac{1}{2}\mathrm{E}_{P_1}\left[-\mathrm{tr}\left(\Sigma_1^{-1}(x-\mu_1)(x-\mu_1)^T\right) + \mathrm{tr}\left(\Sigma_2^{-1}(x-\mu_2)(x-\mu_2)^T\right)\right] \\
&= \frac{1}{2}\log\frac{\det\Sigma_2}{\det\Sigma_1} + \frac{1}{2}\mathrm{E}_{P_1}\left[-\mathrm{tr}\left(\Sigma_1^{-1}\Sigma_1\right) + \mathrm{tr}\left(\Sigma_2^{-1}\left(xx^T - 2x\mu_2^T + \mu_2\mu_2^T\right)\right)\right] \\
&= \frac{1}{2}\log\frac{\det\Sigma_2}{\det\Sigma_1} - \frac{1}{2}n + \frac{1}{2}\mathrm{tr}\left(\Sigma_2^{-1}\left(\Sigma_1 + \mu_1\mu_1^T - 2\mu_2\mu_1^T + \mu_2\mu_2^T\right)\right) \\
&= \frac{1}{2}\left(\log\frac{\det\Sigma_2}{\det\Sigma_1} - n + \mathrm{tr}\left(\Sigma_2^{-1}\Sigma_1\right) + \mathrm{tr}\left(\mu_1^T\Sigma_2^{-1}\mu_1 - 2\mu_1^T\Sigma_2^{-1}\mu_2 + \mu_2^T\Sigma_2^{-1}\mu_2\right)\right) \\
&= \frac{1}{2}\left(\log\frac{\det\Sigma_2}{\det\Sigma_1} - n + \mathrm{tr}\left(\Sigma_2^{-1}\Sigma_1\right) + (\mu_2-\mu_1)^T\Sigma_2^{-1}(\mu_2-\mu_1)\right)
\end{aligned}
$$

Appendix C

Awards

The doctoral study led to the receipt of several best paper awards at top-tier conferences, as depicted in Fig.C.1a-C.1b. The Outstanding Paper Award for Murali et al. (2022a) at the IEEE International Conference on Flexible, Printable Sensors and Systems (FLEPS) 2022 is shown in Fig.C.1a. The paper described the TIQF framework (Chap.4), and its application for active tactile perception. The Outstanding Sensors and Perception Finalist Award for Gentner et al. (2023) presented at the IEEE International Conference on Robotics and Automation (ICRA) 2023 is shown in Fig.C.1b. Although this work is not described in this thesis, it is closely related to the thesis and proposed a robust point cloud registration using the graph-based maximum consensus method, to which contributions were made as a co-author.



(a) *(b)*

Figure C.1: (a) Award certificate: IEEE FLEPS 2022, (b) Award certificate: IEEE ICRA 2023