Efficient Belief Propagation for Perception and Manipulation in Clutter

by

Karthik Desingh

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer Science and Engineering)
in the University of Michigan
2020

Doctoral Committee:

Professor Odest Chadwicke Jenkins, Chair
Professor Dmitry Berenson
Professor Benjamin Kuipers
Professor Edwin Olson
Professor Gaurav Sukhatme, University of Southern California

Karthik Desingh

kdesingh@umich.edu

ORCID iD: 0000-0002-1817-1575

DEDICATION

In the memory of my loving parents,
Desingh
&
Kalaiselvi

ACKNOWLEDGMENTS

First and foremost, I want to express my sincerest gratitude to my advisor, Professor Odest Chadwicke Jenkins, who has been a great advisor, and a wonderful mentor. Without his research vision and utmost patience with my, at times, floundering progress, this thesis would not be possible. Throughout my Ph.D., Chad has provided insightful comments, suggestions, and has consistently pointed out critical problems to be addressed in robotics research. As a mentor, he continuously creates a nurturing environment where one could explore new research ideas, make mistakes, and learn from them. I am indebted to him for his warm and friendly support during many difficult periods over the course of my Ph.D. Thank you, Chad!

I would also like to extend my deepest gratitude to Dmitry Berenson, Benjamin Kuipers, Edwin Olson, and Gaurav Sukhatme for serving on my committee. They have broadened my perspectives with constructive feedback and suggestions. I want to pay special regards to Ben, whose invaluable wisdom and advice helped me figure out purposeful career goals. I would like to sincerely thank Ed for having insightful discussions to clarify my conceptual understanding. I also wish to thank Joseph LaViola from the University of Central Florida for mentoring and collaborating with me on writing research grants. Thanks to the National Science Foundation for funding much of the work developed in this thesis.

The laboratory of PROGRESS at the University of Michigan has given me an inclusive group of friendly, supportive and caring colleagues; Zhiqiang Sui, Zhen Zeng, Zheming Zhou, Kevin French, Jana Pavlasek, Anthony Opipari, Shiyong Lu, Emily Sheetz, Yunwen Zhou, Adrian Röfer, Zhefan Ye, Xiaotong Chen, Thomas Cohn, and Stanley Lewis. I enjoyed our stimulating research discussions and collaborating while sharing a lot of memorable moments during my life as a graduate. Particularly, a special thanks to Zhiqiang and Zhen, whose outstanding research efforts have influenced and motivated some of the works in this dissertation. There is a long list of people, including colleagues, faculty members, and staff from both the University of Michigan and Brown University, who have played an important role in rendering such an amazing graduate education experience.

I want to thank all my dear friends for their unwavering support and love over the years. Especially to Kaushik Vijaykumar, Srinath Ravichandran, and Heather Sousa for painstakingly proofreading several of my manuscripts. Finally, I would like to pay regards to my late parents, who made enormous sacrifices to give me this amazing life. And to my family members for their endless love, support, and encouragement to pursue what truly interests me and for making me the person I am today.

TABLE OF CONTENTS

Dedication . . . . . **ii**

Acknowledgments . . . . . **iii**

List of Figures . . . . . **vi**

List of Tables . . . . . **viii**

Abstract . . . . . **ix**

Chapter

1 Introduction **1**

1.1 Goal-directed Manipulation 3
1.2 Scene Estimation using Generative Inference 5
1.3 Thesis contributions 7

2 Background and Related Work **9**

2.1 Goal-driven Autonomy 9
2.1.1 Perception for Goal-directed Manipulation 10
2.2 Scene Understanding 11
2.2.1 Rigid Body Pose Estimation 12
2.2.2 Parts-Based Recognition 13
2.2.3 Articulated Pose Estimation and Tracking 13
2.3 Foundational Concepts for Generative Inference 14
2.3.1 Importance Sampling 15
2.3.2 Particle Filter 16
2.3.3 Message Passing for Nonparametric Belief Propagation 18
2.4 Summary 22

3 Physics informed Scene Estimation **23**

3.1 Introduction 23
3.2 Related Work 24
3.3 Motivation 25
3.3.1 Object Physical Interactions and Partial Observations 25
3.4 Methods 27
3.4.1 Physics-informed Particle Filter 27
3.4.2 Physics-informed Markov Chain Particle Filter 30

3.5 Experimental Details and Results . . . 31
3.5.1 Iterative Closest Point method . . . 32
3.5.2 Base Clutter Scene Results . . . 33
3.5.3 Cluttered Scene Results . . . 36
3.6 Summary . . . 36

4 Efficient Belief Propagation for Pose Estimation of Articulated Objects . . . 38

4.1 Introduction . . . 38
4.2 Related Work . . . 39
4.3 Method . . . 41
4.3.1 Nonparametric Belief Propagation . . . 41
4.3.2 Pull Message Passing for Nonparametric Belief Propagation . . . 44
4.4 Experimental Details and Results . . . 45
4.4.1 Comparison of Message Passing Algorithms on Articulated Pattern . . . 45
4.4.2 Real World Experiments with RGB-D Observations . . . 48
4.4.3 Articulated Objects Models . . . 50
4.4.4 Baseline . . . 51
4.5 Limitations . . . 56
4.6 Summary . . . 57

5 Belief Propagation for Tracking Pose of Articulated Objects in Clutter . . . 58

5.1 Introduction . . . 58
5.2 Related Work and Background . . . 60
5.3 Methodology . . . 61
5.3.1 Belief Propagation via Message Passing . . . 63
5.4 Object Tracking Experiment . . . 64
5.4.1 Potential Functions for Tracking Experiments . . . 65
5.4.2 Tracking Results . . . 66
5.5 Limitations and Failure Scenarios . . . 70
5.6 Summary . . . 70

6 Conclusion and Future directions . . . 72

6.1 Contributions . . . 72
6.2 Future Directions . . . 73
6.2.1 Learning the Potential Functions . . . 73
6.2.2 Part and Affordance Discovery during Interaction . . . 73

Bibliography . . . 74

LIST OF FIGURES

FIGURE

1.1 General purpose robots and their indoor tasks 2
1.2 Goal-directed Manipulation task 4
1.3 Partial observations in indoor scenes 5

2.1 Importance sampling illustration 15

3.1 Motivational cases for the primitive object interactions 26
3.2 System architecture for physics-informed particle filter (PI-PF) 28
3.3 Results for objects touching experiment 34
3.4 Results for objects stacking experiment 34
3.5 Results for objects slanted experiment 34
3.6 Results for objects occluded experiment 35
3.7 Results for complex experiment with 4 objects 37

4.1 2D articulation pattern and its graphical model 46
4.2 Illustration of unary potential for 2D articulation pattern 46
4.3 Illustration of pairwise sampling for 2D articulation pattern 47
4.4 Convergence in 2D articulation pose under clutter 47
4.5 Convergence in 2D articulation pose under clutter and occlusion 49
4.6 Convergence plot and execution time plot 49
4.7 Error vs execution time plot 50
4.8 Convergence in cabinet pose and comparison with particle filter baseline 51
4.9 Convergence in cabinet pose under different occlusions 52
4.10 Factored pose estimation using PMPNBP extends to articulated objects such as Fetch robot 53
4.11 Cabinet manipulation scenario 1 54
4.12 Cabinet manipulation scenario 2 55

5.1 Accuracy of PMPNBP with respect to the percentage of uniformly sampled belief particles at every iteration 59
5.2 Message passing via augmentation and selection steps 61
5.3 Augmentation Illustration for clamp's top part 62
5.4 Convergence characteristics of different selection methods 64
5.5 Inference pipeline for tracking the pose of articulated objects 65
5.6 Tracking experiment with occlusion under no interaction 67

5.7 Tracking experiment with occlusion during interaction 68
5.8 Tracking experiment with background clutter 68
5.9 Tracking experiment with task demonstration 69
5.10 Tracking experiment showing the limitation of the tracking pipeline 70

LIST OF TABLES

TABLE

3.1 Object pose estimation errors are reported here with respect to the ground truth poses. Ground truth is generated by manually matching the object geometries to the observed point cloud using the Blender user interface. In all the experimental categories (touching, stacked, slant and occluded), physical informed estimators PI-PF, PI-MCPF perform better than the ICP method. The variance of the physics informed methods are higher in the slant cases as the simulations result in different plausible slant pose every time. In the occluded category of experiments, the ICP method has NA entries as the method is not applicable when no sensor data is available. 33

3.2 Shows the average number of iterations each of methods, took to converge. Maximum iterations are the number of iterations each method is allowed to run. We consider the experiment to have converged if the change in the pose estimate of the most likely particle is less than 1 cm in position and less than 3 degrees in the angles. 37

ABSTRACT

Autonomous service robots are required to perform tasks in common human indoor environments. To achieve goals associated with these tasks, the robot should continually perceive, reason its environment, and plan to manipulate objects, which we term as goal-directed manipulation. Perception remains the most challenging aspect of all stages, as common indoor environments typically pose problems in recognizing objects under inherent occlusions with physical interactions among themselves. Despite recent progress in the field of robot perception, accommodating perceptual uncertainty due to partial observations remains challenging and needs to be addressed to achieve the desired autonomy.

In this dissertation, we address the problem of perception under uncertainty for robot manipulation in cluttered environments using generative inference methods. Specifically, we aim to enable robots to perceive partially observable environments by maintaining an approximate probability distribution as a belief over possible scene hypotheses. This belief representation captures uncertainty resulting from inter-object occlusions and physical interactions, which are inherently present in clutterred indoor environments. The research efforts presented in this thesis are towards developing appropriate state representations and inference techniques to generate and maintain such belief over contextually plausible scene states. We focus on providing the following features to generative inference while addressing the challenges due to occlusions: 1) generating and maintaining plausible scene hypotheses, 2) reducing the inference search space that typically grows exponentially with respect to the number of objects in a scene, 3) preserving scene hypotheses over continual observations.

In order to generate and maintain plausible scene hypotheses, we propose physics informed

scene estimation methods that combine a Newtonian physics engine within a particle based generative inference framework. The proposed variants of our method with and without a Monte Carlo step showed promising results on generating and maintaining plausible hypotheses under complete occlusions. We show that estimating such scenarios would not be possible by the commonly adopted 3D registration methods without the notion of a physical context that our method provides.

To scale up the context informed inference to accommodate a larger number of objects, we describe a factorization of scene state into object and object-parts to perform collaborative particle based inference. This resulted in the Pull Message Passing for Nonparametric Belief Propagation (PMPNBP) algorithm that caters to the demands of the high-dimensional multimodal nature of cluttered scenes while being computationally tractable. We demonstrate that PMPNBP is orders of magnitude faster than the state-of-the-art Nonparametric Belief Propagation method. Additionally, we show that PMPNBP successfully estimates poses of articulated objects under various simulated occlusion scenarios.

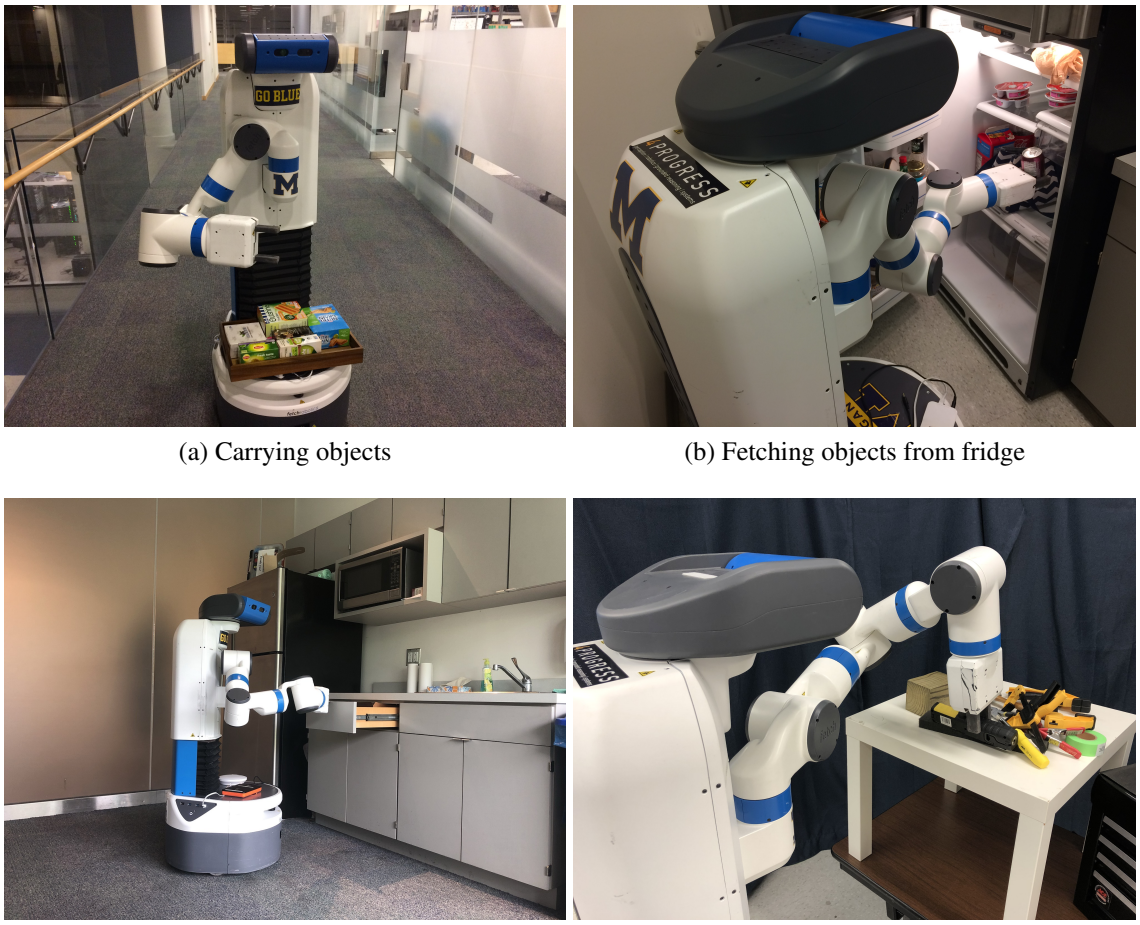
To extend our PMPNBP algorithm for tracking object states over continuous observations, we explore ways to propose and preserve hypotheses effectively over time. This resulted in an augmentation-selection method, where hypotheses are drawn from various proposals followed by the selection of a subset using PMPNBP that explained the current state of the objects. We discuss and analyze our augmentation-selection method with its counterparts in belief propagation literature. Furthermore, we develop an inference pipeline for pose estimation and tracking of articulated objects in clutter. In this pipeline, the message passing module with augmentation-selection method is informed by segmentation heatmaps from a trained neural network. In our experiments, we show that our proposed pipeline is able to effectively maintain belief and track articulated objects over a sequence of observations under occlusion. We show that the efficient nonparametric belief propagation (PMPNBP) proposed in this dissertation can be effectively applied to solve perceptual problems in robotics where a notion of uncertainty due to partial observations is inevitable.

CHAPTER 1

Introduction

Autonomous service robots have the potential to assist humans with day-to-day tasks in indoor environments. These robots should be capable of doing all the activities one would like a butler to do in a typical household environment. However, in reality, our robots cannot see the world as humans do. We have had tremendous success in making robots function efficiently where the environment is structured, such as warehouse autonomous navigation or industrial manufacturing. The imposed structure complements the inability of the robots to perceive their environments. However, our household environments are highly unstructured, inherently complex with a variety of objects, interactions and relations, and associations to indoor locations. Consider the tasks highlighted in Figure 1.1, such as carrying objects in the house, fetching objects from a fridge, organizing your kitchen by placing objects into drawers, and working with tools. To perform any of such tasks, the robot butler should know what objects are involved, where they are, and how to grasp and move them around to accomplish the task. With a wide range of objects in the indoor environments and limited onboard sensing systems, the desired robotic perception under unstructured indoor environments is challenging and largely unsolved. To achieve this long term goal of having a general-purpose robot with the capabilities of a butler, the problem of perception under complex unstructured indoor environments should be addressed.

In addition to perceiving complex environments, for a robot to interact fluidly with human partners, it must be able to interpret scenes in the context of a human's model of the world. Humans ground their perception with high-level reasoning and express their model of the world in the form of language symbols. This ability to ground the symbols that conceptually tie low-level perception with high-level reasoning is a critical missing component for a robot to possess autonomy. We specifically face the problem of anchoring, i.e. to ground the symbols and associate physical objects in the real world and their relationships, with computationally assertable facts via robot perception. Once the robot is capable of grounding the symbols to represent a state of the world, it needs to perform high-level reasoning over a sequence of actions to achieve a task-oriented goal. With this capability in robots, human users will be able to more intuitively specify goals for robots, as desired states of the world semantically [1].



(a) Carrying objects

(b) Fetching objects from fridge

(c) Placing objects into drawers

(d) Working with tools

Figure 1.1: General purpose robots and their indoor tasks: A general purpose autonomous robots should possess capabilities of robustly interacting with daily objects in indoor environments and perform various tasks

Classical sequential planning algorithms that provide high-level reasoning over a sequence of actions to achieve a task-oriented goal have over a five-decade history [2, 3]. With axiomatic representations of the world as a requirement, a classical planner makes an unrealistic assumption that the physical robot has a full perception of the environment. The robot's sensing and action in the real world are dominated by uncertainty. This uncertainty is a result of both sensor measurements and motor actuation of the robot. For example, sensor measurements are frequently not adequate to identify and localize occluded or partially visible objects. The resulting noisy and incomplete descriptions of a state subsequently affecting the axiomatic representation of the world and are unsuitable inputs for existing planning algorithms.

Generative models provide a means to address uncertainty probabilistically. These models generate and maintain a distribution of possible hypotheses to explain the sensory observations instead of discriminating the state of the world. These generated hypotheses form an approximate probability distribution (or belief) over possible states of the world. A state estimate from the resulting

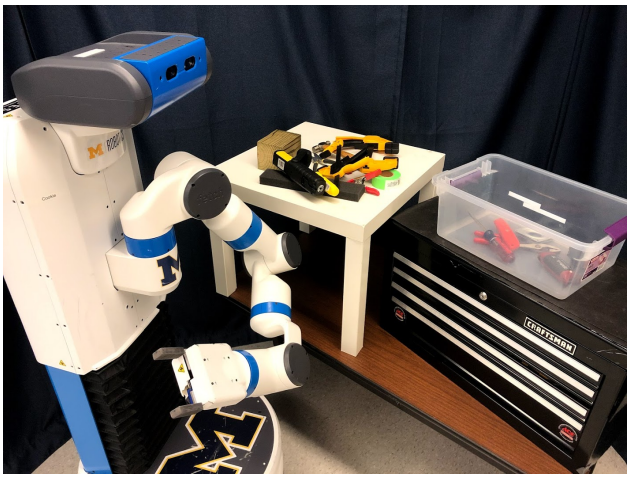
belief distribution represents the current state of the world for classical planning, thus avoiding the intractability of planning in the space of this belief. This decoupled approach to maintain the distribution on the perceptual side while using an estimate toward planning is ubiquitously used in the autonomous navigation for planning from localization state estimates.

In this thesis, we propose generative methods to enable a robot to perform goal-directed manipulation. Specifically, we propose particle-based inference methods to perceive complex indoor environments under partial observations, modeled as probabilistic graphical models. The remainder of this chapter introduces to the concept of goal-directed manipulation with a motivating example, description of scene estimation using generative inference, and a high-level overview of the thesis' contributions.

1.1 Goal-directed Manipulation

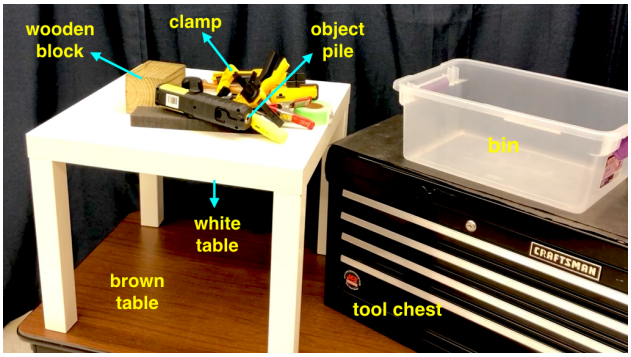
Imagine the near future, where the Fetch robot (mobile manipulation platform at the University of Michigan) helps a carpenter in his workspace. Consider the scenario shown in Figure. 1.2(a), where the robot is encountering a scene with a pile of tools and objects on a *white table*, beside a *tool chest* and a *bin*. Assume that the robot's task is to first put away the tools on the *white table* (Figure. 1.2(b)) into the *bin*, and then to clamp a *wooden block* using a *bar clamp* to the *white table*. Figure. 1.2(c) shows the desired goal scene that has a *wooden block* clamped to the *white table* while all the other tools and objects either in the *tool chest* or in the *bin*. With this goal configuration associated to the task at hand, the robot needs to perceive: 1) the *object pile* on the *white table* 2) the *bin* and 3) the *tool chest*, and plan a sequence of actions towards the goal and execute the next best action at every time step. Each action leads to a new world state, which is either the predicted outcome of the action or different due to uncertainty in the previously perceived world state or the action performed. The robot has to perceive the resulting new world state and inform the planner to plan the next action towards the goal. This continual process of perception, planning, and action execution is performed until the robot achieves the desired goal configuration.

Let us assume that we have complete observability of the world, and the robot can perceive it and represent it symbolically. The arbitrary initial scene (Figure. 1.2(b)) and the goal scene (Figure. 1.2(c)) of our example scenario can be symbolically represented using relational scene graphs. A relational scene graph represents the world state as a directed graph with nodes denoting objects and edges representing the contact relations such as "on", "in", "holds" and "contact". For example, the *white table* is "on" the *brown table* in captured in Figure. 1.2(c & d). By representing the current state and the goal state symbolically, classical planners such as STRIPS [2] and SHRDLU [3] can be used to plan a sequence of high-level actions that will lead to intermediate states directed

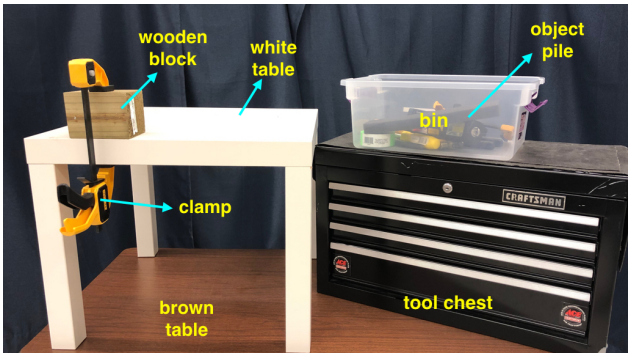


(a) A robot observing a workspace - a possible scenario in the real world

(b) An object pile on a white table (from the view of the robot) - an example of how real world objects could be cluttered



(c) An arbitrary initial scene



(d) Desired goal scene

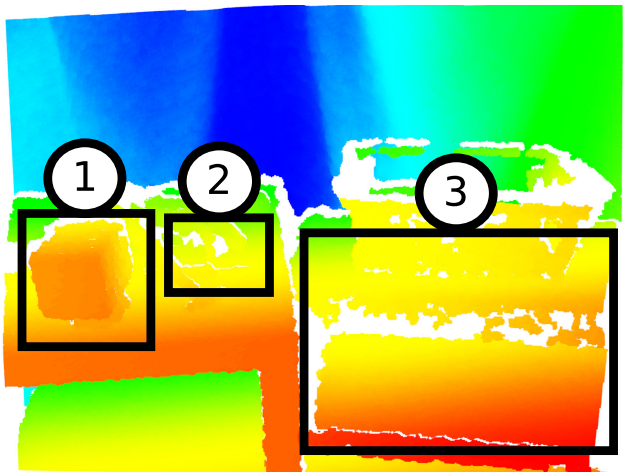
Figure 1.2: Goal-directed Manipulation task: Robot observing an arbitrary scene with a desire to achieve the goal configuration in order to accomplish a task

towards the desired goal state. In addition to the symbolic representation of the scene for the task planner, the perception system should also provide the metric 6D pose (position and orientation) of the objects involved in the high-level action. Following the decision of a high-level action from a planner, the inverse kinematics and motion planning algorithms [4, 5] use the 6D poses of the target objects (for example an object from the *object pile* and the *bin*) to let the robot manipulator reach the joint arm configurations that execute an action such as pick and place.

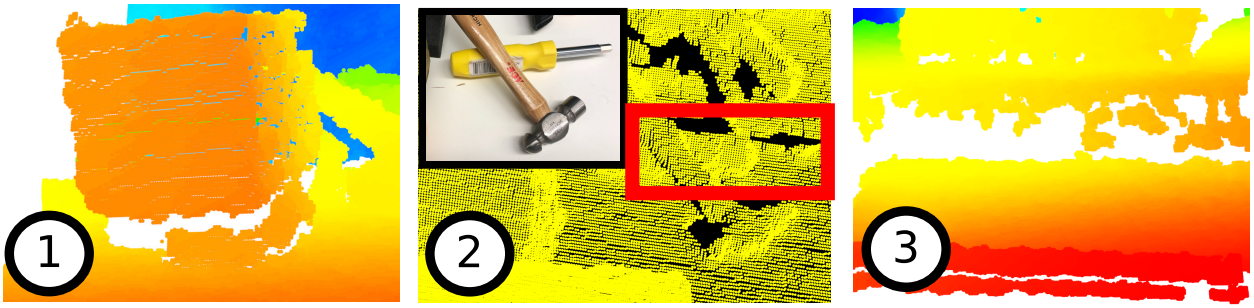
However, the assumption of complete observability using raw sensor observations is unrealistic, given the complexity of a typical human environment that can only be partially observed by a robot. The key factors contributing to the challenge in perceiving indoor environments are partial sensor observations due to:

- occlusions resulting due to physical interactions between objects,

(a) Original scene observed by the robot



(b) 3D point cloud from the sensor



(c) Highlighted cases of physical interaction (1), physical properties of an object (2) and self-occlusion of articulated object (3), and partial observations obtained

Figure 1.3: Partial observations in indoor scenes

- self and environmental occlusions due to jointed objects that have parts that move,
- differences in size, shape, appearance, and material properties of the objects.

In addition to these factors, the search space for estimating the world as a collection of objects, their 6D poses, with their inter-object relations, is a multi-modal high-dimensional continuous state space. Perceiving the world is a challenging problem that needs to be addressed in order to enable robots to perform goal-directed manipulation tasks. This thesis addresses the challenges in perceiving objects under uncertainty due to partial sensor observations, in the context of goal-directed manipulation in clutter environments.

1.2 Scene Estimation using Generative Inference

A perception system for goal-directed manipulation should account for the key factors contributing to the partial sensor observations mentioned in the previous subsection. Figure. 1.3, shows three

cases where the Fetch robot is observing the world using its 3D sensor that produces 3D point cloud data. In each of these cases, the observed 3D point cloud data has only partial or incomplete information resulted due to physical interactions between objects, their geometrical properties, and articulations of their jointed parts. Case (1) shows a scenario where, a *wooden block* is supported by a *tape measure*, creating only a partial observation of the *tape measure* in the 3D point cloud data. Case (2) shows a scenario where the small metallic part of the screwdriver is only captured partially by the sensor because of its size and material properties. Case (3) shows a *tool chest* in a configuration, where the topmost drawer is open, occluding the rest of the drawers resulting in a partial observation of the entire object. Scene inference methods should handle all these partial observation scenarios and inform a planner to drive the goal-directed manipulation.

In addition to the above scenarios, partial observations arise from a limited view of the sensor, manipulation actions, or both. For instance, while picking objects from the white table, the robot will have a limited view of the tool chest, resulting in the partial or incomplete observation of the tool chest and its state in the world. Similarly, during the action execution, the robot may obstruct its view of the scene. These scenarios are ubiquitous during the entire task execution, necessitating the ability to perceive the world through sensor observations that vary over time. Hence, scene inference methods should not only estimate the poses of the objects but also track them over time by maintaining a distribution of possible state hypotheses of the world.

Scene inference involves detection and pose estimation of objects in the scene. Object detection identifies what objects are in the scene while pose estimation provides their 6D poses in the world. The 6D pose that encapsulates position and orientation is defined with respect to object geometries (3D mesh models). The scene observed by the robot can be explained by performing detection and pose estimation collectively for all objects. The current literature in the field of object detection and pose estimation can be broadly classified as either being discriminative [6, 7, 8, 9, 10, 11] or generative [12, 13, 14]. Discriminative methods are powerful in extracting features that could drive inference problems. However, hard thresholding on object classes and poses may provide incorrect information for goal-oriented robotic tasks and not limited to single manipulation action. Comparatively, generative methods are slow (given current computational limitations), but can explain the plausible composition of a scene and produce a belief space. The latter approach is suitable for goal-directed manipulation because of three main reasons:

- Partial sensor observations (as shown in Figure. 1.3) can be explained by a generative approach to provide the robot with a distribution over possible hypotheses (belief), where each hypothesis is a scene state with a collection of objects poses.

- Generated belief can be propagated over changing observations overtime under action execution.

- Goal-directed manipulation involves a sequence of actions perturbing or changing the world in the process. Hence, a belief generated before the first action can be updated by the predicted state outcome after the action and corrected by the resulting new observation. This process enables the robot to recover from uncertainty in perception and actuation.

The generative approaches to solving traditional robotic state estimation problems such as localization, mapping and Simultaneous Localization and Mapping (SLAM), has resulted in great success leading to working physical systems in the autonomous navigation domain. We take a similar approach and perform belief space scene inference and demonstrate its benefits for goal-directed manipulation.

Object pose in the real world gives the position and orientation of the object, which gives a set of relative poses for the robot's gripper to reach in order to accomplish a high-level action such as pick and place. Going back to the motivating scenario as shown in Figure. 1.2, the high-level actions could be categorized and sequenced as 1) picking objects other than *wooden block* and *clamp*, from the table and dropping them into the bin, 2) picking tools other than the *wooden block* and the *clamp*, from the table and placing them inside the *tool_chest* after opening the appropriate drawer, 3) manipulating the *clamp* such that the clamping action could be performed. Apart from the pick and drop scenario, the robot should have the notion of the pose of the objects to achieve the placement and clamping action. Task-specific grasping and manipulation necessitate the notion of a pose to object geometries which, combined with relative poses of the robot's gripper, provides a variety of actions that an object affords.

1.3 Thesis contributions

In pursuit of enabling robust goal-directed manipulation capabilities, this thesis aims to address the problems of perception under uncertainty in cluttered environments using generative inference methods. More formally, the problem statement for this dissertation is as below.

Problem statement: How to estimate a scene as a collection of rigid body poses, while efficiently maintaining and propagating a distribution over possible hypotheses, under robot's partial sensor observations.

The specific contributions of each of the chapters are described below.

- Chapter 2 provides the relevant background to the thesis, with the current standing of the robotics research community on enabling a robot to perform sequential manipulation tasks. Additionally, this chapter also provides the theoretical concepts that are fundamental to the particle-based inference methods proposed in chapters 3-5.

- Chapter 3 aims at estimating a cluttered scene with objects under physical interactions. We propose a particle-based inference method that generatively estimates a scene ensuring physical plausibility. Specifically, we propose two variants of the particle-based inference framework [15] that uses Monte Carlo sampling approaches. The developed algorithms explain partial observations with objects under heavy occlusions by producing estimates that are plausible in the real world.

- Chapter 4 aims at making the generative scene inference tractable using factorization of a scene into objects and object-parts. This factorization is formulated as a Markov Random Field (MRF) and solved using Nonparametric Belief Propagation. We propose fast inference using a Pull Message Passing algorithm (PMPNBP) [16, 17] and demonstrate its efficiency by estimating scenes with articulated objects. We show that our proposed method has a significant gain in computation compared to a state-of-the-art message passing algorithm.

- Chapter 5 aims to tackle the problem of pose estimation and tracking of scene hypotheses, where a scene is composed of articulated objects under partial time-variant observations. Specifically, this chapter explores the sum-product and max-product variants of belief propagation algorithms while catering to the needs of the tracking problem. We developed a framework that utilizes a segmentation neural network to inform the message passing module about an object part's appearance. We demonstrate that the proposed framework can maintain belief over the pose of an object articulated during a human demonstration, and thus track the object over continuous observations even under occlusions.

CHAPTER 2

Background and Related Work

In this chapter, we survey the related work in the field of goal-driven autonomy in robotics, specifically goal-directed manipulation that motivates the objectives of this dissertation. We provide relevant works in the literature that focus on perceptual problems in the context of grasping and manipulation. Additionally, we briefly describe theoretical concepts that are fundamental to understanding the inference methods proposed in chapters 3-5.

2.1 Goal-driven Autonomy

Goal-driven autonomy is the desired capability of an autonomous robot to robustly perform a sequence of actions and accomplish a task-oriented goal. Additionally, to interact fluidly with human partners, a robot must interpret scenes in the context of a human user's model of the world. The challenge is that many aspects of the human world model are difficult or impossible for the robot to sense directly. We posit that the critical missing component is the grounding of symbols that conceptually tie low-level perception and high-level reasoning for extended goal-driven autonomy. We specifically face the problem of anchoring [18], a case of symbol grounding [19], to associate physical objects in the real world and relationships between these objects with computationally assertable facts (or axioms), from the robot's perception of the world. Anchoring and symbol grounding are at the heart of the emerging area of semantic mapping [20] and its accelerated growth due to advancements in 3D RGBD mapping [21, 22]. With a working memory of grounded axioms about the world, robot manipulators will be able to flexibly and autonomously perform goal-directed tasks that require reasoning over sequential actions. Just as important, human users will be able to more intuitively specify goals for robots, as desired states of the world, through spatial configurations. Human users should be able to program the robots in an intuitive way to communicate the tasks, which is termed as *Robot Programming*.

Existing research in the direction of programming robots approached learning low-level skills from users in the form of demonstrations. Different approaches have been proposed in Program-

ming by Demonstration (PbD) for low-level learning of skills, such as trajectories [23, 24] and control policy [25, 26] in robot configuration space. These methods are inherently limited to world states in a workspace that is similar to the ones in the demonstrations. By representing the goal of a task in the workspace instead of in the configuration space, goal-directed autonomy can reason and plan its actions to reach the goal from arbitrary initial world states. Other work has focused on the high-level aspects of a task. Veeraraghavan et al. [27] propose learning a high-level action plan for a repetitive ball collection task from demonstrations. Ekvall et al. [10] focus on learning task goals and use a task planner to reach the goal. Chao et al. [28] provide an interface for the user to teach task goals in a tabletop workspace. However, these methods simplify the scene perception problem using planar objects, box-like objects, or objects with distinguishing colors, that are far from real-world scenarios. Yang et al. [29] have proposed learning action plans in real-world scenarios. This dissertation enables robot programming from a human user perspective, especially on real-world tasks with real-world objects, by specifying or just showing desired states of the world.

2.1.1 Perception for Goal-directed Manipulation

We term goal-driven autonomy with actions limited to grasping and manipulation actions as Goal-directed Manipulation. In the context of manipulation, we aim to estimate axiomatic representations of the world that will allow robotics to build on the body of work in sequential planning algorithms, such as STRIPS [2] and SHRDLU [3]. A classical planner can compute actions for a robot to perform arbitrary sequential tasks assuming full perception of the environment, which is often an unrealistic assumption. Nevertheless, in structured perceivable environments, systems based on classical planning have demonstrated the ability to perform goal-directed manipulation reliably. Mohan et al. [30, 31] uses the Soar cognitive architecture with axiomatic scene graph representation [32] for teaching a robot arm to play games such as tic-tac-toe, Connect-4, and Towers of Hanoi through language-based expressions. Chao et al. [28] perform taskable symbolic goal-directed manipulation with a focus on associating observed robot percepts with knowledge categories. This method uses background subtraction to adaptively build appearance models of objects and obtain percepts but with sensitivity to lighting and object color. Narayanaswamy et al. [33] perform scene estimation and goal-directed robot manipulation for cluttered scenes of toy parts for the flexible assembly of structures.

Tenorth and Beetz [34] developed the KnowRob system to performs taskable goal-directed sequential manipulation at the scale of buildings by automatically synthesizing information from the semantic web and Internet. The KnowRob system focuses uncertainty at the symbolic level and relies on hard and complete state estimates from hardcoded software components [35]. Similarly, Srivastava et al. [36] perform the joint task and motion planning, taking advantage of modifications

in controlled environments, which include green screens and augmented reality tags.

While domains with uncertainty are traditionally problematic for classical planning, we posit that advances in robot perception and manipulation with new approaches to anchoring can overcome this uncertainty for goal-directed robot control. There have been several discriminative methods proposed to perceive exact single estimates of scene state for manipulation, which both complement and inspire our probabilistic methods in this dissertation. Based on the semantic mapping work of Rusu et al. [37], the canonical manipulation baseline is the PR2 Interactive Manipulation pipeline [38]. This pipeline can perform relatively reliable pick-and-place manipulation for non-touching objects in flat tabletop settings. This pipeline relies upon the estimation of the largest flat surface, by clustering of computed surface normals. Any contiguous mass of points extruding from this support surface is considered a single object, leading to many false positives in object recognition and pose estimation.

Rosman and Ramamoorthy [39] address such point cloud segmentation issues in relational scene graph estimation by detecting contact points between objects from depth observations. Collet et al. [40] propose a system for recognition and pose registration of common household objects from a single image by using local feature descriptors. Papazov et al. [41] perform sequential pick-and-place manipulation using a bottom-up approach of matching known 3D object geometries to point clouds using RANSAC and retrieval by hashing methods. Cosgun et al. [42] present a novel algorithm for placing objects by performing a sequence of manipulation actions in cluttered surfaces like the tabletop. ten Pas and Platt [43], and Mahler et al. [44] proposed object agnostic grasp localization methods in highly unstructured scenes of diverse objects. Joho et al. [45] use a generative model to cluster objects on a flat surface into semantically meaningful categories. Similarly, Dogar et al. [46, 47] consider active manipulation of highly occluded non-touching objects on flat surfaces. In contrast to these methods, the methods proposed in this dissertation focus on maintaining a distribution over all possible scenes, and not reliant upon selecting and maintaining a hard (potentially incorrect) state estimate for perception.

2.2 Scene Understanding

Scene inference constitutes to the estimation of a scene as a collection of all the objects. Specific to the tasks involving grasping and manipulation, object localization in the 3D workspace is a common component in scene inference. An object's location and orientation in the scene is represented with the notion of pose associated with the object geometry. Estimating such a pose has received considerable attention in robotics. In this thesis, we focus on determining a scene as a collection of objects, their rigid body parts under articulation. Additionally, this dissertation also explores the ability to extend the inference methods to track objects continually over time. In relevance

to the thesis, we discuss the related work that focuses on rigid body pose estimation, parts-based recognition, articulated object estimation, and object tracking.

2.2.1 Rigid Body Pose Estimation

We can categorize the conventional approaches to estimate a rigid body pose into three: 1) generative, 2) discriminative, 3) generative-discriminative inference methods.

Generative inference provides a means to address uncertainty probabilistically and robust to noisy observations. Generatively possible world states can be hypothesized to explain the true world state that could have generated the robot's observations. These generated hypotheses form an approximate probability distribution (or belief) over possible states of the world. Zhang and Trinkle [48] formulated a physics-informed particle filter, Grasping-Simultaneous Localization, and Modeling, and Manipulation (G-SLAM) for grasp acquisition in occluded scenes. Sui et al [49, 13] proposed a generative inference method to estimate axiomatic scene graphs. Similarly, we propose a physically plausible scene estimation method [15] (described in Chapter 3) that uses a physics engine within a particle-based generative inference framework. Zhou et al. [50] proposed a generative method to localize objects under layered translucency using plenoptic (light-field) observations. While these generative methods are robust to noisy observations, they are computationally expensive.

Discriminative methods, on the other hand, are computationally efficient and have a faster recall power. Recently discriminative methods using end-to-end learning frameworks are proposed to estimate the 6D pose of the objects. Xiang et al. propose an end-to-end network for estimating 6D pose from RGB images [6]. This work was further extended to make use of synthetic data generation and augmentation techniques to improve performance [51]. Wang et al. [52] propose an end-to-end network that uses depth information along with RGB to estimate the pose. These methods rely significantly on good texture and are constrained to a dataset of objects. Also, estimates from the discriminative methods are noisy and less reliable, especially in challenging cluttered scenarios.

Combining the discriminative power of feature-based methods with generative inference has been successful under challenging conditions such as background and foreground clutter [53, 54, 55], adversarial environment conditions [56] as well as uncertainty due to robot actions [57, 1]. The success of the above approaches inspires us to utilize the speed of discriminative methods to inform our generative inference methods (described in Chapter 5).

2.2.2 Parts-Based Recognition

Parts-based representations have been proposed to aid scene understanding, and action execution [58, 59, 60], and have recently garnered attention within the robotics and perception communities [61, 62]. Parts-based localization has led to research in recognizing objects and their articulated parts [63]. Parts-based perception for objects in human environments is often limited to recognition and classification tasks. Parts-based pose estimation is often considered for human body pose [14], and hand pose [64] estimation problems with fixed graphical models. In this thesis, we factor a scene or object, into its rigid body parts to accommodate challenging cluttered scenarios. To address the computational limitations of existing belief propagation algorithms, we propose an efficient message passing algorithm [17]. We demonstrate its utility on articulated object pose estimation (see Chapter 4), and extend it further to a tracking framework (see Chapter 5).

2.2.3 Articulated Pose Estimation and Tracking

Probabilistic modeling has been widely applied to object tracking. Wuthrich et al. [65] propose a probabilistic technique for tracking of objects being manipulated by a human or robot with known geometries using a particle filter. The particle filter models occlusions alongside the observation and process models. The framework was extended to track a manipulator end-effector [66]. In [67], Schmidt et al. introduce a general framework for tracking articulated objects with the known articulation using an extended Kalman filter, where the observation model employs the signed distance function. It was extended to include physics-based constraints on the objects [68]. Makris et al. [69] propose a hierarchical model fusion framework for visual tracking in which a defined object model hierarchy guides the inference of the main model by fusing the inferences made on simpler auxiliary models. Issac et al. [70] modify the Gaussian filter to track object models robustly and efficiently. These tracking frameworks are either initialized to objects' ground truth poses or informed by joint encoder readings in the case of articulated objects. Full scene estimation and tracking of known objects were studied in the context of SLAM by Salas-Moreno et al. [71]. However, this work assumes objects have no articulations and are static while the camera is in motion. In this thesis, we aim to develop a unified framework that performs pose estimation and tracking of articulated objects without any initialization (see Chapter 4 and 5).

In the existing literature, a particular focus has been placed on addressing the task of estimating the kinematic models of articulated objects by a robot through interactive perception [72]. Hausman et al. [73] propose a particle filtering approach to estimate articulation models and plan actions that reduce model uncertainty. In [74], Martin et al. suggest an online interactive perception technique for estimating kinematic models by incorporating low-level point tracking and mid-level rigid body tracking with a high-level kinematic model estimation over time. Sturm et

al. [75, 76] addressed the task of estimating articulation models in a probabilistic fashion by a human demonstration of manipulation examples. Using the articulation models produced by these earlier works, in this dissertation, we address the problem of estimating and tracking their poses under challenging partial observations.

Li et al. [77] explore category level localization of articulated bodies in a point cloud. However, their method does not consider clutter and occlusions from the environment. Michel et al. [78] perform one-shot pose estimation of articulated bodies using 3D correspondences with optimization over hypotheses. All of these above approaches consider large, primarily planar objects that cover a significant portion of the observation as opposed to tools and small objects in this work.

2.3 Foundational Concepts for Generative Inference

In this section, we describe the foundations of Importance Sampling (IS), and how nonparametric methods such as Particle Filter (PF), and Nonparametric Belief Propagation (NBP) are derived from IS. This dissertation aims to develop inference methods for continuous, high-dimensional, and multi-modal random variables that represent complex indoor environments in robot perception problems. Nonparametric generative methods such as PF and NBP can perform inference of random variables under considerable uncertainty and are well suited to tackle the aforementioned problems. However, the direct application of existing algorithmic instances of these methods is computationally expensive. This dissertation explores ways to overcome this computational cost, and proposes algorithms with the goal of attaining tractable inference on robot perception problems. Here, we provide sufficient background on the existing nonparametric methods and their respective algorithmic instances in order to understand the efficient instances proposed in chapters 3-5.

Derived from the principles of Importance Sampling, PF and NBP are designed for solving problems with different graphical models. PF is effective in solving problems that are modeled as a Markov chain, where the hidden variables have a chain-like decomposition which capture temporal concepts. However, for a given instance in time, perception problems are often modeled by complex graphical models containing high–dimensional variables. Because particle filtering cannot be applied directly to arbitrary graphs, a nonparametric version of belief propagation was developed (Sudderth et al. [79] and Isard et al. [80]). NBP is catered to general graph structures with non-Gaussian edge potentials.

As stated by Isard et al. [80], NBP is Belief Propagation on particle networks where the inference is performed by iteratively passing messages between the hidden nodes. Various message passing algorithms have been proposed over the years to cater to address the particular needs of each application. Based on the objective of the inference task, they are broadly classified into

sum-product (SP) and max-product (MP) algorithms. SP variants are used for marginal inference, whereas the MP variants are used for Maximum *a posteriori* (MAP) inference. In this section, we describe the fundamental differences between these two categories of algorithms specific to NBP, and their approximations specific to cyclic graphs, while discussing some of the existing methods in the literature.

2.3.1 Importance Sampling

Sampling algorithms are aimed at generating set of samples, or particles, that represent the underlying distribution of a random variable. This particle set (sample-based representation) has the ability to model any arbitrary distribution given that the number of particles is large enough. It is often not feasible to sample from an arbitrary distribution. There are various sampling techniques in the literature that make use of a known density in order to sample efficiently, but then weight these samples according to the arbitrary target distribution, to recover a more representative set of samples. Importance Sampling (IS) is one such technique that is commonly used. Here, we describe IS and its variant with a *resampling step* described in Probabilistic Robotics [81].

IS is a technique that is used to sample from a *target* density function $f(X)$ using another density function generally referred to as a *proposal* function $\pi(X)$. This technique is used when there is no direct means of sampling from the *target* density function. Each drawn sample from the *proposal* density is given a weight $w_i = \frac{f(X_i)}{g(X_i)}$, which represents the mismatch in the density values of the two distributions $f(X)$, and $\pi(X)$ (see Figure. 2.1).

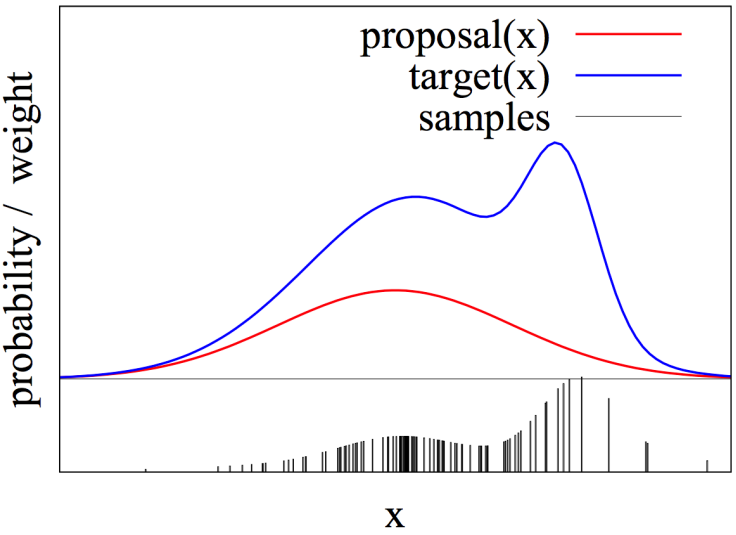


Figure 2.1: Importance sampling illustration: The samples are drawn from π and reweighted by $\frac{f(X)}{\pi(X)}$ to produce weighted samples that represent $f(X)$. This figure is from Stachniss 2006 [82]

The problem addressed by Important Sampling is in computing an expectation $E_f[X]$ over a

probability density function f, given the samples from the proposal distribution π. The transformation utilized by IS (restated from Todkar et al [83]) is shown below.

$$E_f[X] = \int f(x)dx$$
$$E_f[X] = \int \frac{f(x)}{\pi(x)}\pi(x)dx$$
$$E_f[X] = E_\pi[\frac{f(X)}{\pi(X)}X]$$

2.3.2 Particle Filter

In many Monte Carlo problems, X is a high-dimensional random variable and the target density $f(X)$ induces a chain-like decomposition of X. This decomposition paves the way to generate X sequentially as $X_{1:t}$, where t is the time. IS specific to such problems is known as Sequential Importance Sampling (SIS) (Todkar et al [83] for other variants of Importance sampling). SIS is designed for state-space models that have a chain-like structure. If f is written as

$$f(X) = f(X_1)\prod_{t=2}^{d} f(X_t|X_{t-1}), \tag{2.1}$$

an importance sampling scheme can be built using π as

$$\pi(X) = \pi_1(X_1)\prod_{t=2}^{d} \pi_t(X_t|X_{t-1}), \tag{2.2}$$

where density of $\pi_t(X_t|X_{t-1})$, mimics the intermediate target density $f(X_t|X_{t-1})$. The importance weight in this case can be computed sequentially to obtain $w(X) = w_t$, where

$$w_t = w_{t-1}\frac{f(X_t|X_{1:t-1})}{\pi_t(X_t|X_{1:t-1})} \tag{2.3}$$

and $w_0 = 1$. Equation 2.3, can be further decomposed as

$$w_t = w_{t-1}\frac{f_t(X_{1:t})}{f_{t-1}(X_{1:t-1})\pi_t(X_t|X_{1:t-1})}. \tag{2.4}$$

Here, the densities $f_t(X_{1:t})$ could be chosen to approximate the marginal densities $f(X_{1:t})$. $f_t(X_t|X_{1:t-1})$ guides the proposal density $\pi_t(X_t|X_{1:t-1})$. We refer to Section 3 of Liu 2001 [84] for more details.

Applying SIS to interesting problems with high-dimensional state variables proved to be inefficient due to a collapse of distinct samples of the target density $f(X)$. Gordon et al. [85], proposed

a *resampling* step to SIS, to account for this collapse. This ad-hoc step improved the algorithm's effectiveness on state-estimation problems, by retaining important and distinct samples. This variant of SIS with resampling proposed by Gordon et al. [85] is commonly known as a *particle filter* or *bootstrap-filter*, and is described in Probabilistic Robotics by Thrun et al. [81].

In the context of robot state estimation, the target density $f(X)$ is the posterior over all states $p(X_{0:t}|U_{1:t}, Z_{1:t})$. X_t, U_t and Z_t denote the state, action taken, and observation at time t by the robot respectively. As derived in Probabilistic Robotics by Thrun et al. [81] (see section 4.3.3),

$$p(X_{0:t}|U_{1:t}, Z_{1:t}) = \eta\, p(Z_t|X_t)p(X_t|X_{t-1}, U_t)p(X_{0:t-1}|Z_{1:t-1}, U_{1:t-1}). \tag{2.5}$$

With $p(X_{0:t}|U_{1:t}, Z_{1:t})$ being equivalent to $f_t(X_{1:t})$, and $p(X_t|X_{t-1}, U_t)$ being equivalent to $f_t(X_t|X_{1:t-1})$, Equation 2.4 specific to the robot state estimation can be rewritten as

$$w_t = w_{t-1}\frac{p(X_{0:t}|U_{1:t}, Z_{1:t})}{p(X_{0:t-1}|U_{1:t-1}, Z_{1:t-1})p(X_t|X_{t-1}, U_t)}. \tag{2.6}$$

Using Equation 2.5, this further simplifies to

$$w_t = \eta\, p(Z_t|X_t). \tag{2.7}$$

As described in the particle filter algorithm (see Table 4.3 in Probabilistic Robotics [81]), M samples are drawn from the proposal $p(X_t|X_{t-1}, U_t)$. More specifically, $X_t^{[m]} \sim p(X_t|X_{t-1}^{[m]}, U_t)$ and weighted as $w_t^{[m]} = p(Z_t|X_t^{[m]})$. These M samples are resampled using their weights. This "trick", often referred to as a *resampling* step, carries forward duplicate samples from the previous set to the new set, therefore retaining important and distinct samples. This "trick" of *resampling* is some times loosely referred to as *importance sampling* (or even *importance resampling*) itself (see Probabilistic Robotics [81] section 4.3.3).

In Chapter 3, we propose scene estimation methods (Axiomatic Scene Estimation [49] and Physics informed Scene Estimation [15]) that are adapted from the particle filter. In our methods, we perform static scene estimation and assume that there is no physical action performed by the robot, and hence U_t is an identity function. In order to avoid the particle sets collapsing to a few identical and distinct samples as a result of the *resampling* step, we add Gaussian noise to the samples. It can be argued that this proposal function in the form of Gaussian noise is of a non-probabilistic nature and breaks the underlying assumptions made in the particle filter or Bayesian filter as described in Probabilistic Robotics [81]. However, as stated earlier, the proposal $\pi_t(X_t|X_{t-1})$ should mimic the intermediate target density $f_t(X_t|X_{t-1})$, which is the intention of the Gaussian proposal. Hence, the scene inference methods proposed in [15, 49] broadly fall un-

der SIS. Additionally, there exist similarities in how the methods [15, 49] compare to the *particle swarm optimization* [86] that optimizes by iteratively improving the population of candidate solutions.

2.3.3 Message Passing for Nonparametric Belief Propagation

Perception problems are often modeled by complex graphical models resulting in inference of continuous, high-dimensional, and multi-modal random variables. Because the particle filter cannot be applied directly to arbitrary graphs, Sudderth et al. [79] and Isard et al. [80], developed Nonparametric Belief Propagation (NBP). In this formulation, the problem is treated as a network of particle filters performing local inference while maintaining global consistency.

The problem is formulated as a Markov Random Field (MRF) with graph $G = (V, E)$, vertices $s \in V$, edges $(s, t) \in E$, and density:

$$P(X) \propto \prod_{s \in V} \phi_s(X_s) \prod_{(s,t) \in E} \psi_{st}(X_s, X_t), \tag{2.8}$$

where ϕ_s and ψ_{st} are unary and pairwise clique potentials respectively. As described in [87], consider a simple acyclic graph that produces a joint probability as a product of cliques

$$p(X_1, X_2, X_3, X_4) \propto \\ \phi_1(X_1)\phi_2(X_2)\phi_4(X_3)\phi_4(X_4)\psi_{12}(X_1, X_2)\psi_{23}(X_2, X_3)\psi_{24}(X_2, X_4) \tag{2.9}$$

To compute $p(X_1)$, we can marginalize the joint probability from Equation. 2.9 as

$$p(X_1) \propto \phi_1(X_1) \\ \underbrace{\int \phi_2(X_2) \overbrace{\left(\int \phi_3(X_3)\psi_{23}(X_2, X_3)dX_3 \right)}^{m_{3\to 2}(X_2)} \overbrace{\left(\int \phi_4(X_4)\psi_{24}(X_2, X_4)dX_4 \right)}^{m_{4\to 2}(X_2)} dX_2}_{m_{2\to 1}(X_1)}. \tag{2.10}$$

Belief Propagation (BP) provides a methodical way of computing the marginals in Equation 2.10 for every hidden node. For a Markov Random Field (MRF) the local belief $bel_s(X_s)$ (marginals $p(X_s)$) and messages $m_{t\to s}(X_s)$ (from t to s) are given by:

$$bel_s(X_s) \propto \phi_s(X_s) \prod_{t \in \rho(s)} m_{t\to s}(X_s) \tag{2.11}$$

$$m_{t\to s}(X_s) = \int_{\mathcal{X}_t} \phi_t(X_t)\psi_{st}(X_s, X_t) \prod_{u\in\rho(t)\setminus s} m_{u\to t}(X_t)dX_t \tag{2.12}$$

The marginal $p(X_s)$ over X_s is given by the product of messages $m_{t\to s}(X_s)$ from neighbors $t \in \rho(s)$, and the local evidence $\phi_s(X_s)$. The message from node t to s is computed recursively by multiplying incoming messages to node X_t with the local evidence $\phi_t(X_t)$ and compatibility potentials $\psi_{st}(X_s, X_t)$, and then integrating over X_t.

2.3.3.1 Sum-Product variant of BP

The process of computing marginals via local message recursions is popularly categorized as sum-product (SP) variant of BP, as the updates involve products over messages, and for discrete models the integrals become summations.

For discrete models, Equation 2.12 changes to

$$m_{t\to s}(X_s) = \sum_{X_t\in\mathcal{X}_t} \phi_t(X_t)\psi_{st}(X_s, X_t) \prod_{u\in\rho(t)\setminus s} m_{u\to t}(X_t). \tag{2.13}$$

When the domain $\mathcal{X}_s$ is too large, a sample based representation can be sought after, resulting in Particle Belief propagation (PBP) [88]. The message $m_{t\to s}(X_s)$ is an expectation of samples drawn using a proposal $W_t(X_t)$

$$m_{t\to s}(X_s) = E_{X_t\sim W_t}\left[\psi_{st}(X_s, X_t)\frac{\phi_t(X_t)}{W_t(X_t)} \prod_{u\in\rho(t)\setminus s} m_{u\to t}(X_t)\right]. \tag{2.14}$$

In other words, for a particle $X_s^{(i)}$ with index i, the message $m_{t\to s}(X_s^{(i)})$ gives us the corresponding weight over the particle set $\mathbb{X}_t$ drawn using the proposal $W_t(X_t)$. Given a sample set $\mathbb{X}_t = \{X_t^{(1)}, ..., X_t^{(n)}\}$ drawn from $W_t(X_t)$, we can estimate $m_{t\to s}(X_s^{(i)})$ using Importance Sampling (IS) to give

$$\hat{m}_{t\to s}(X_s^{(i)}) = \frac{1}{n}\sum_{j=1}^{n} \psi_{st}(X_s^{(i)}, X_t^{(j)})\frac{\phi_t(X_t^{(j)})}{W_t(X_t^{(j)})} \prod_{u\in\rho(t)\setminus s} \hat{m}_{u\to t}(X_t^{(j)}). \tag{2.15}$$

Ihler et al. [88] proved that Equation 2.15 agrees with Equation 2.13 as $n \to \infty$. Further, they provide theoretical guarantees for the convergence properties of particle based representations for BP and show that the convergence rate is $\mathcal{O}(\frac{1}{\sqrt{n}})$. In Chapter 4, we group the methods from Ihler et al. [88] (PBP), Sudderth et al. [79] (NBP), and Isard et al. [80] (PAMPAS) under the name of Nonparametric Belief Propagation (NBP) as all of them are sample-based instances of

Belief Propagation (BP) with the objective of estimating marginal expectations and, additionally all belong to the category of sum-product variants.

We are motivated by the earlier works [14, 64] that apply NBP to articulated body pose estimation and tracking problems. NBP methods [79, 80] provided sampling approaches to perform belief propagation with continuous variables. Both approaches approximated a continuous function as a Gaussian mixture and used local Gibbs sampling to approximate the product of mixtures. However, these methods can hardly be generalized to three-dimensional (3D) articulated pose estimation problems because of their high computational expense. In the dissertation, we propose a more efficient "pull" message passing algorithm for nonparametric belief propagation (PMPNBP) [16, 17] (described in Chapter 4). The key idea of pull message updating is to evaluate samples taken from the belief of the receiving node with respect to the densities informing the sending node. The approximation of mixture products can be performed individually per sample and then normalized into a distribution. PMPNBP avoids the computational pitfalls of "push" updating used in NBP [79, 80], and show applicability for 3D articulated pose estimation with compelling examples. PMPNBP falls into the category of sum-product variants and has similarities to PBP (Ihler et al. [88]) when it comes to particle representation. PBP emphasizes the advantages of using a large number of particles to represent incoming messages, along with theoretical analysis. This work uses an expensive iterative Markov Chain Monte Carlo sampling step, mimicking the Gibbs sampling step in other NBP approaches [79, 80]. PMPNBP is able to avoid this cost through a resampling step. Specifically, our complexity is $\mathcal{O}(DM)$ in computing a message mixture of M components using D incoming mixtures as compared with $\mathcal{O}(DKM^2)$ of NBP [79, 80, 88] with K product sampling iterations.

2.3.3.2 Max-Product variant of BP

Another objective in statistical inference is to quantify uncertainty about the maximizing configuration of random variables, known as Maximum *a posteriori* (MAP) inference. MAP is suitable for applications where a jointly consistent estimator is preferred. The max-product (MP) variant of BP is similar to the sum-product form, where messages maximize, instead of marginalizing over, joint state. The standard max-product algorithm is similar to sum-product BP, but the integration in Equation 2.13 is replaced by a maximization over all $X_t \in \mathcal{X}_t$ to give

$$\hat{m}_{t\to s}(X_s) = \max_{X_t \in \mathbb{X}_t} \phi_t(X_t)\psi_{st}(X_s, X_t) \prod_{u \in \rho(t)\setminus s} \hat{m}_{u\to t}(X_t) \tag{2.16}$$

where $\mathbb{X}_t$ is the particle set in $\mathcal{X}_t$ such that $\mathbb{X}_t \subset \mathcal{X}_t$. The marginals obtained using these max-product messages are known as *max-marginals* [89, 90]. The notion of max-marginals is introduced in Wainwright et al. [89], for their utility in computing Maximum *a posteriori* probability

(MAP) estimates on tree structures and cyclic graphs. Let $\mu_s(X_s)$ be the max-marginal of X_s defined as

$$\mu_s(X_s) = \max_{X'|X'_s=X_s} p(X'). \tag{2.17}$$

This max-marginal is computed using the messages from Equation 2.16, using the equation:

$$\mu_s(X_s) \propto \phi_s(X_s) \prod_{t\in\rho(s)} \hat{m}_{t\to s}(X_s) \tag{2.18}$$

For tree-structured graphs, the above max-marginals are exact [89]. For cyclic graphs, the max-marginals computed are approximate and are termed as *pseudo-max-marginals* [89]. To compute the MAP-configuration or the joint state estimate X' such that $X' \in \text{argmax}_{X'}\, p(X')$, a backtracking procedure using these max-marginals can be used [90].

In the context of applying the belief propagation to continuous, high-dimensional inference problems, especially for articulated object pose tracking, sum-product variants have been proposed and effectively used [64, 14]. Pacheco et al. [91] proposed a max-product variant called Diverse Particle Message Product (D-PMP) and applied it to the articulated human pose estimation problem. The motivation for D-PMP is that when there are multiple possible modes (more than one person in the observation), sum-product variants face particle degeneracy and collapse to a single mode. However, by adopting the max-product message updates with an optimization routine, D-PMP can maintain diversity in the particle set representing the max-marginals. Their method D-PMP outperforms other variants of max-product such as Metropolis Particle Max-Product (M-PMP) [92], Greedy Particle Max-Product (G-PMP) [93] and PatchMatch & Top-N Particle Max-Product (T-PMP) [94] in estimating multiple articulated structures in the observation.

In Chapter 5, in order to extend PMPNBP toward a tracking framework, we are inspired by an augmentation step used in D-PMP. Additionally, we conduct an experiment to compare the sum-product (PMPNBP) with the max-product (D-PMP) in estimating a single articulated pattern in the observation. Our analysis shows that both PMPNBP (with augmentation), and our implementation of D-PMP have similar covergence properties, i.e. the error in the estimation respect to the number of iterations. To compute the error in the estimation, a single estimate is derived from these marginals and max-marginals, using a post processing step. In other words, the marginals from the PMPNBP, and max-marginals from the D-PMP are used to construct a joint estimate via this post processing step. In our experiments, this post processing step at the end of every iteration, mimics the max-product message update to produces a single MAP estimate. The sum-product variant (PMPNBP) is computationally efficient with complexity of $\mathcal{O}(D\alpha M)$ as compared to the max-product variant (D-PMP) with complexity of $\mathcal{O}(D\alpha M^2)$, where αM is the size of augmented particle set.

2.4 Summary

In this chapter, we surveyed the related work in the field of goal-driven autonomy in robotics, specifically in the field of perception for goal-directed manipulation, which motivates the objectives of this dissertation. We briefly described theoretical concepts such as Importance Sampling, Particle Filter and Message passing for Nonparametric Belief propagation, that are fundamental to understanding the inference methods proposed in chapters 3-5.

CHAPTER 3

Physics informed Scene Estimation

3.1 Introduction

Inferring a cluttered scene with objects under a pile as shown in the motivating scenario (Figure. 1.2(b)), is fraught with challenges, such as occlusions and physical contacts. These challenges prevent acceptable levels of scene perception and, consequently, manipulation and task completion. Even when object geometries are known, the estimation of even a single object is a challenge addressed by recent research [95]. The challenge for scene perception becomes much greater as the scene becomes more cluttered, with an increasing number of objects. A common approach for tabletop scenes is to assume objects are physically separated [96], essentially removing the challenge of clutter.

Addressing this challenge for cluttered environments, we posit that physical plausibility is a necessary component in the estimation of scenes for robot manipulation. The challenges of perception in cluttered scenes are caused directly by the physical configuration and interactions between objects, as well as partial observability from the robot's viewpoint. As with similar analogous approaches to human tracking [97, 98], respecting physical viability often provides improved accuracy in the presence of uncertainty and efficiency in disregarding implausible scene configurations. For example, consider a case of a robot looking down at a large object stacked on top of a (completely occluded) small object. Current methods often misinterpret this scene as a single large box floating above the support surface. In addition to floating objects, physically implausible scene estimates can also occur due to inter-penetrating objects, unsupported objects, and unstable structures.

In this chapter, we propose a means for incorporating physical plausibility into generative probabilistic scene estimation using Newtonian physical simulation. Assuming geometry (dimensions), friction, and mass properties of N unique objects in 3D as known parameters, we explore three approaches to inference as a form of physics-informed scene estimation for static environments. In each of these methods, we use a physical simulation engine to constrain inference to the set of

physically plausible scene states, which we treat as a *physical plausibility projection*. In terms of Bayesian filtering, we describe a physics-informed particle filter (PI-PF) that uses physical plausibility projection to correct implausibility that can occur due to additive diffusion. Based on the idea of [99], we bring PI-PF and MCMC sampling techniques together as a physics-informed Markov Chain Particle Filter (PI-MCPF), where MCMC is performed within the resampling stage of the particle filter.

We provide ICP based approaches as the baseline, to discuss the limitations of data-driven approaches and the advantage of the proposed methods. We present results for inference with the three physics-informed state estimators in primitive cases of cluttered scenes with two objects and more complex scenes with three and four object cases. While our results suggest that the PI-PF and PI-MCPF produce comparable estimation results, we observed the PI-MCPF converges in fewer iterations, albeit with more computational cost per iteration. Using our physics-informed estimators, we demonstrate manipulation of cluttered scenes with a PR2 robot.

3.2 Related Work

Physical context plays a key role in human visual perception, where physical laws inform us that objects are always supported and cannot float and interpenerate. Embodied systems such as robot should have this understanding when interacting and changing the environment during sequential task execution. The problem addressed by our physics informed Bayesian inference in this is to infer object-level manipulation semantics from 3D point clouds, or 3D maps more generally. In terms of utilizing physics, Dogar et al. [100] have incorporated quasi-physical prediction for grasping heavily-occluded non-touching objects cluttered on flat surfaces.

In terms of generative inference, there has been considerable work in using physics within Bayesian filtering models for tracking of people [101, 98] often for locomotion-related activities. Such physics-informed tracking applied to manipulation scenes presents new challenges as the complexity of several interacting objects introduces more complex contact and occlusion dynamics. Outside of robotics and manipulation, recent work by Wu et al. [102] estimated the physical properties of an object using physics engine with deep learning techniques over an input video. This shows recent interest in using Newtonian physics for perceptual tasks. Work by Jia et al. [103] used physics stability to improve the RGBD-segmentation of objects in clutter that could eventually be used to estimate 3D geometry for manipulation. However their physics stability is not done over 3D geometries and precision of their method to suit robotic applications such as manipulation is unknown. Liu et al. [104] used knowledge-supervised MCMC to generate abstract scene graphs of the scene from 6D pose estimates from uncertain low level measurements. Joho et al. [105] used Dirichlet process to reason about object constellations in a scene, helping

unsupervised scene segmentation and completion of a partial scene. Tejas et al. [106] showed that discriminative approaches aid generative models in analysis by synthesis framework to solve scene perception problems, where synthesis takes physical properties into account. Zhang et al. [107] formulated a physics-informed particle filter, G-SLAM, for grasp acquisition in occluded planar scenes. Sui et al. [49] proposed a similar model for estimating the entire relational scene graph and object pose demonstrated relatively small scenes with simple geometries. Narayanan et al. [108] have similar assumptions as ours and formulated the object localization task under occlusions as a multi-hueristic search problem to search over the space of hypothesized scenes. Collet et al. [109] proposed MOPED framework that uses iterative feature clustering for object recognition and pose estimation, and heavily relies on visual features. The methods above are often restricted to quite simplistic scenes and do not consider physical interaction between objects like we do.

In this chapter, we address these challenges by focusing on specific cases of inter-object interaction for estimating the object pose across all six degrees-of-freedom for each object. Distinguishing our work from above methods, we substantiate the accuracy of the object pose estimation by performing robotic manipulation task on the estimated scenes.

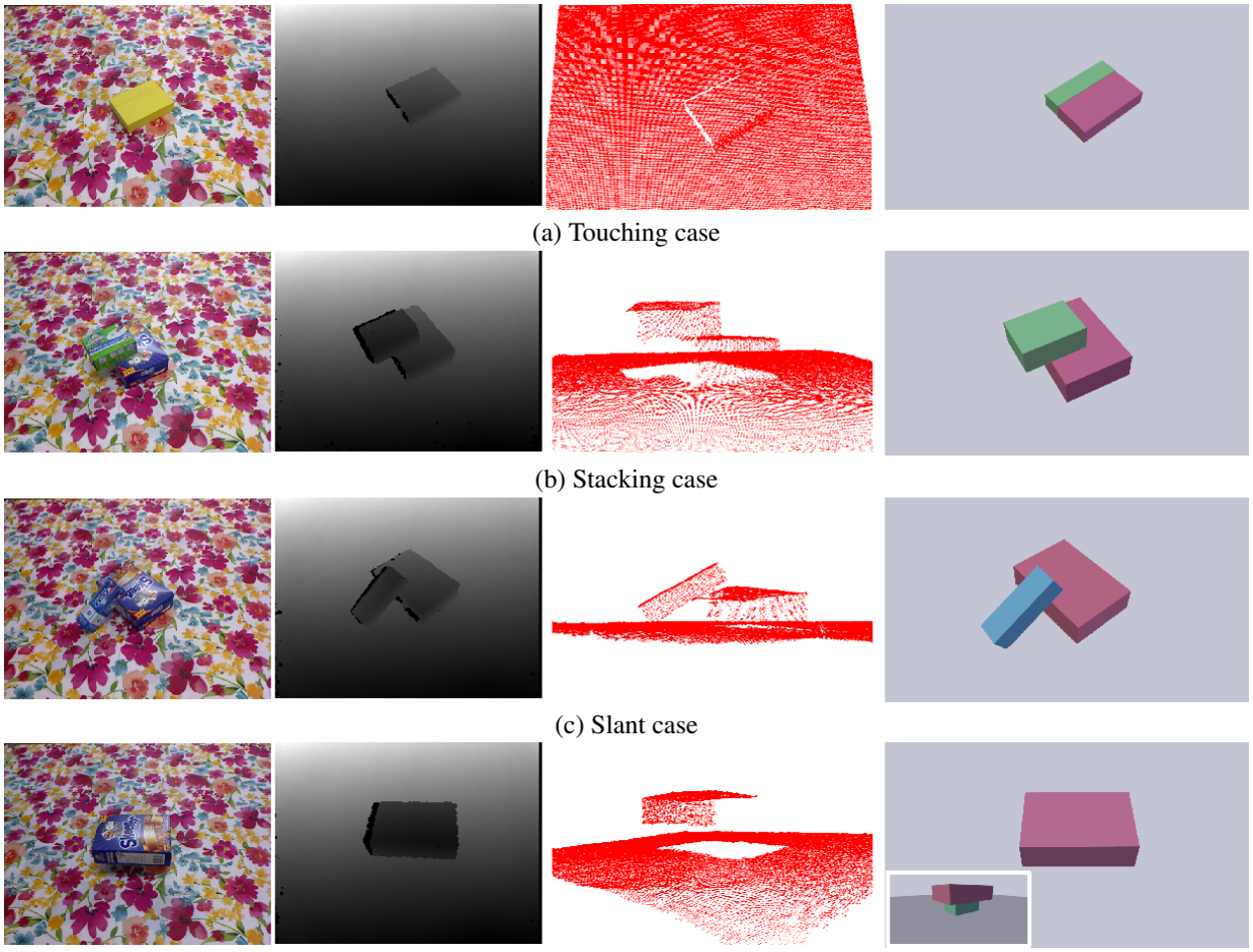
3.3 Motivation

A cluttered scene can be defined as a scene where objects are not segregated from each other and, as a result, not optimally visible to a sensor. Because robotic applications demand reasonable precision in perception to perform even a simple pickup task, the complexity multiplies as the number of objects grow, leading to an increasingly cluttered scene. There are a vast number of object interactions that can cause a scene to be cluttered with this growth in objects. For now, we consider the form of the uncertainty caused by object interactions, and not issues of clutter that might arise with number of objects. As such, we review here the primitive cases of cluttered from physical object interaction: a) objects touching each other, b) objects stacked on top of each other, c) slant objects supported by either their edge or face and, d) objects completely occluded from view by other objects. General clutter scenes are some combination of these four cases.

3.3.1 Object Physical Interactions and Partial Observations

3.3.1.1 Object touching

Consider a case where two objects touch, as shown in Fig. 5.1 (a), with similar texture and appearance. From the depth sensing, these objects could be segmented as a single cluster of objects from the tabletop. However, there is no discriminable depth discontinuity between the objects.



(a) Touching case

(b) Stacking case

(c) Slant case

(d) Occlusion case

Figure 3.1: Motivational cases for the primitive object interactions, commonly seen in cluttered scenes: Left to right, Real world scene, depth observation, point cloud view and estimated scene (using our approach) in blender view

Under-constrained and discriminative methods that depend on features, such as corners or pre-segmentation, often fail to estimate the touching cases reliably. Our proposal to use a generative approach can be advantageous in these scenarios as shown in Fig. 5.1 (a).

3.3.1.2 Object stacking

Another frequent interaction between objects is stacking. Consider a two object stacking case as shown in Fig. 5.1(b), where the top object is close to the edge of the bottom object. The depth data as seen in the point cloud view of Fig. 5.1(b) is very sparse. RGBD feature extraction and/or discrimination might be able to detect the objects in the scene but precise pose estimation would still be a problem as it will depend on the sparse depth data observed. Further, an ambiguous pose estimation might lead to states that are not physically plausible. For example, an estimate could have poses with the center of the mass of the top object away from the edge of the bottom object,

towards unsupported space. This results in a state estimate that is not plausible with the physics of the environment. Therefore, we claim that integrating physics as a part of the estimation process is essential to reject such implausible hypotheses and converge to the ground truth scene as shown in the Fig. 5.1(b).

3.3.1.3 Object slant

Cluttered scenes may also include piles of objects, which produces cases where objects are not just supported by one of their faces, but by their edges and corners. Consider two objects slant case as shown in Fig. 5.1(c), where one object is oriented such that its mass is supported both by the table and the other object. With the sparse depth data as shown in the Fig. 5.1(c), pose estimation of the slant objects is challenging. In addition, a wrong estimation of the pose of the slant object might lead to objects inter-penetrating. Our proposed method is able to handle the slant object cases which requires consideration of an object‘s possible inter-penetration and its physically plausible constraints.

3.3.1.4 Object occlusion

Object occlusion is another common problem in cluttered scenes; it ranges from partial occlusion to complete occlusion of objects. Consider two objects as shown in Fig. 5.1(d), where one object is on top of a second object that is not visible to the sensor. This configuration results in the data driven approaches being unaware of the bottom object, unless a prior informs of the bottom object being at a known location. A generative approach, such as ours, hypothesizes object poses that produce scenes matching to the observation shown in Fig. 5.1(d). Occluded objects will have multiple pose hypotheses that generate scenes to best match the observation. Our Bayesian filter approach maintains a distribution over these possible poses and estimates the likely pose of the occluded object in the next time frame when the scene is acted upon by a robot.

3.4 Methods

3.4.1 Physics-informed Particle Filter

We denote our physics-informed particle filter as PI-PF. We model this problem of pose estimation as a recursive Bayesian filter, a common model used for state estimation in robotics [110]. The Bayesian filter is described by the following equation, with X_t being the state of the scene X at

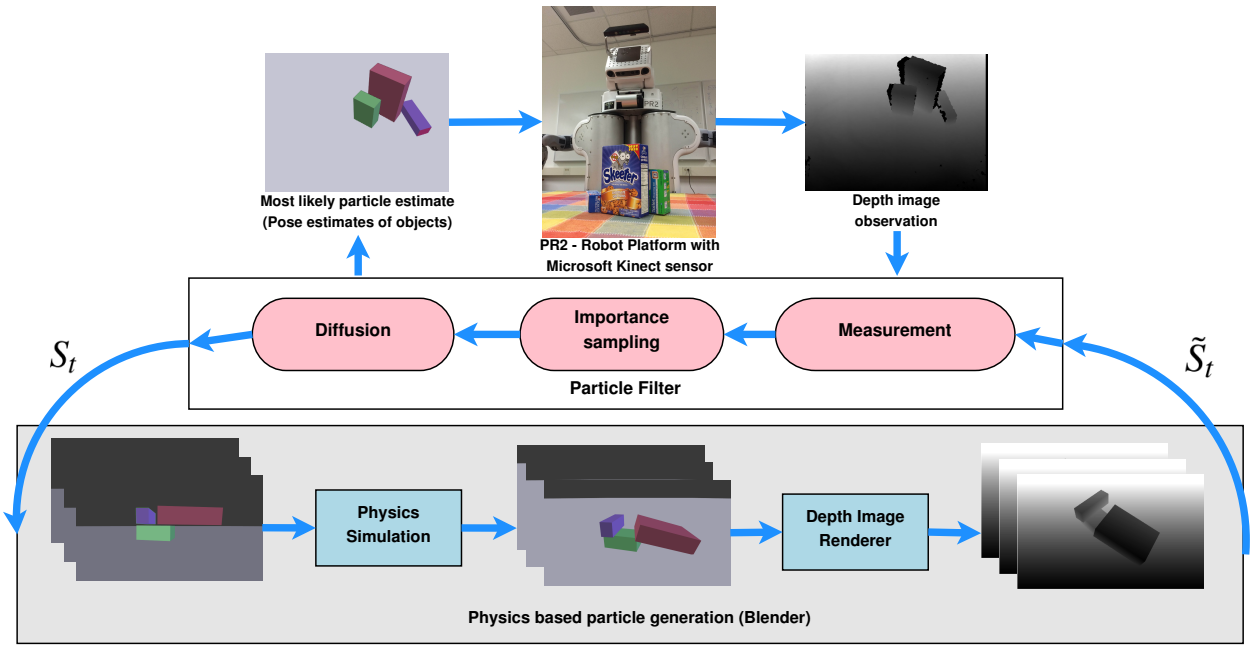


Figure 3.2: System architecture for physics-informed particle filter (PI-PF)), for viable pose estimation of objects: Robot observes the scene as a depth image and infers the state by a particle filter approach, where each particle is a hypothesized scene rendered by a graphics engine followed by a physics projection to ensure its plausibility in the real world. After iterating for a set of particles with measurement update and diffusion, the most likely particle is estimated to be the state of the scene.

time t, sensory observations Z_t, control actions U_t taken by the robot:

$$p(X_t|Z_{1:t}) \propto p(Z_t|X_t) \int p(X_t|X_{t-1}, U_t) p(X_{t-1}|Z_{1:t-1}) \mathrm{d}X_{t-1}. \tag{3.1}$$

Scene state X_t is a set of object poses in the scene, represented as $X_t = \{p_1, p_2, p_3, ...p_m\}$. Pose of an i^{th} object in a scene state is $p_i = \{x_i, y_i, z_i, \varphi_i, \theta_i, \psi_i\}$ where x_i, y_i, z_i are the 3D position of the center of mass and φ_i, θ_i, ψ_i are three Euler angles parameterizing the rotation in space. $S_t = \left\{X_t^1, X_t^2, X_t^3, ...X_t^N\right\}$ represents a set of scenes or particles before physics plausibility projection. $\tilde{S}_t = \left\{\tilde{X}_t^1, \tilde{X}_t^2, \tilde{X}_t^3, ...\tilde{X}_t^N\right\}$ represents a set of scenes or particles after physics plausibility projection. U_t is the sum of the user forces applied to the set of objects, which will be zero for this current work.

Our proposed framework consists of two major components: a particle filter and the physics based particle generator (Fig. 3.2). Initially, a set of n particles is generated randomly (uninformed by the observation) to form S_t states. Each particle X_t^j is physically projected to a state $\tilde{X}_t^j$ and thus forms $\tilde{S}_t$ set of states. The particle filter consists of *measurement*, *importance sampling* and *diffusion* submodules. The *measurement module* takes in the observation Z_t in the form of depth image given by the robot's depth sensor and physically viable particles $\tilde{S}_t$ generated by the physics based particle generator (a set of depth images rendered by a 3D z-buffer renderer). The *measure-*

ment module compares each of the particle $\tilde{X}_t^j$ represented as depth image with the observation Z_t using sum squared distance function over every pixel. This comparison gives the likelihood of each particle being close to the observation. The *importance sampling* module takes the likelihood of all the particles to perform resampling of states, based on their likelihood. This process generates more particles created with the states that were plausible. These states are diffused by the *diffusion* submodule to provide the states for the next iteration S_t.

It should be noted here that the states S_t generated by the *diffusion module* are not guaranteed to be physically viable. Therefore, physics based particle generator takes the states produced after the diffusion from the filter and projects it to $\tilde{S}_t$. These projected states are then rendered out as depth images and the process continues till the convergence is reached.

As alluded to above, the sequential Bayesian filter in Eq. 4.1 is commonly approximated by a collection of N weighted particles, $\{X_t^{(j)}, w_t^{(j)}\}_{j=1}^N$, with weight $w_t^{(j)}$ for particle $X_t^{(j)}$, expressed as:

$$p(X_t|Z_{1:t}) \propto p(Z_t|X_t) \sum_j w_{t-1}^{(j)} p(X_t|X_{t-1}^{(j)}, U_{t-1}) \tag{3.2}$$

From this approximation, we will still resample as in standard particle filtering by drawing N updated samples:

$$X_t^{(j)} \sim \pi(X_t|X_{t-1}^{(j)}, U_{t-1}). \tag{3.3}$$

Because $X_t^{(j)}$ are potentially physically implausible, we will apply a function f to each of these drawn samples to produce a new set of physically-plausible particle hypotheses:

$$\tilde{X}_t^{(j)} = f(X_t^{(j)}, V_t^{(j)}, h). \tag{3.4}$$

where $f(X_t^{(j)}, V_t^{(j)}, h)$ is the function integrating a model of Newtonian physics forward in time by h seconds from the positions $X_t^{(j)}$ and velocities $V_t^{(j)}$ of objects in a scene. Because we are considering static scenes, it should be noted that both the object velocities $V_t^{(j)}$ and control forces U_t are assumed to be zero in magnitude. The resulting set of physically-viable particles are used to form an approximation of the posterior at time t by computing the new weights $\tilde{w}_t^{(j)}$ through evaluating their likelihood:

$$\tilde{w}_t^{(j)} = p(Z_t|\tilde{X}_t^{(j)}), \tag{3.5}$$

and normalizing to sum to one:

$$w_t^{(j)} = \frac{\tilde{w}_t^{(j)}}{\sum_k \tilde{w}_t^{(j)}}. \tag{3.6}$$

Although we are considering static scenes, it should also be noted that the particle filter is able to perform tracking over time for moving objects as well with non-null object velocities and control

forces.

With regard to function f, given the geometry of a rigid object and its physical properties (mass, inertia and friction), a stable position and orientation of this object can be computed with gravitational and contact forces using a physics simulator. We cast *physical plausibility projection*, as the process of submitting a state X_t^j of the scene, which might not be physically plausible or stable, as an initial condition of the physics simulator in order to generate a guaranteed physically plausible and stable state $\tilde{X}_t^j$ at the end of the simulation.

An example of physics projection is shown in the Fig. 3.2. The scene state from the diffusion module is not guaranteed to be physically stable. As shown in Fig. 3.2, the green object is stable on the surface, whereas the other two objects are floating in the air. When a scene goes through the physical simulation, it is projected to a state that is physically stable as shown in the Fig. 3.2. This projection could lead to stacking and slant cases as in this example where the blue object is stacked on top of green and the red object rests in a slant position supported by the green object. There are many other physically implausible cases such as object inter-penetrations and center of mass not fully supported by other objects in the scene, that can be projected to a stable scene with this physics projection. These examples show how physics brings realism to the estimation process, making it a plausible perception.

3.4.2 Physics-informed Markov Chain Particle Filter

We explored Markov Chain Monte Carlo (MCMC) [111], a popular method employed for inference in scene estimation problems. To integrate physically stable sampling strategy into the single-site Metropolis Hastings algorithm [111], every new sample X^* generated from proposal distribution $q(X_t^*|\tilde{X}_{t-1})$ has to be physically projected, where $\tilde{X}_{t-1}$ is the previous sample. The proposal distribution $q(X_t^*|\tilde{X}_{t-1})$ is defined as a $\mathcal{N}(\tilde{X}_{t-1}, \Sigma)$, where Σ is the same as used in the diffusion of PI-PF. It should be noted that the generated sample X_t^* is not guaranteed to be a physically plausible state. Hence, we project the X_t^* to $\tilde{X}^*$ using function f as shown in Eq 4.4.

The physics projection of the new sample makes the random walk in the neighborhood no more a useful sampling technique. Hence, we discarded the direct application of MCMC method with physics plausibility check and instead integrate MCMC in our PI-PF method to improve the posterior distribution represented by the collection of the particles. This method of inference is inspired by Khan et al. [99] for MCMC in particle filter for tracking. Once we have $\tilde{S}_t$, a set of physically viable particles in PI-PF at iteration t, we sample a different particle as proposed by $q(X^{*(j)}|\tilde{X}_t^{(j)})$ to get $S_t^* = \left\{X_t^{*1}, X_t^{*2}, X_t^{*3}, ...X_t^{*N}\right\}$. S_t^* is then physically projected to get $\tilde{S}_t^* = \left\{\tilde{X}_t^{*1}, \tilde{X}_t^{*2}, \tilde{X}_t^{*3}, ...\tilde{X}_t^{*N}\right\}$. Now, an acceptance probability check is performed on each particle $\tilde{X}_t^{*(j)}$, to either accept or reject each of these new samples to get a new set $\tilde{S}_t$ for the

iteration t. The acceptance probability check is defined as below.

$$A(\tilde{X}_{t-1}^{(j)}, \tilde{X}_t^{*(j)}) = min\Big\{1, \frac{L(\tilde{X}_t^{*(j)})}{L(\tilde{X}_{t-1}^{(j)})}\Big\}. \tag{3.7}$$

where $L(X_t)$ is the likelihood of a state X_t given by the below equation.

$$L(X_t) = p(Z_t|X_t) \tag{3.8}$$

When $A(\tilde{X}_{t-1}^{(j)}, \tilde{X}_t^{*(j)})$ is 1, then the new sample $\tilde{X}_t^{*(j)}$ is accepted to be $\tilde{X}_t^j$, else a random number α from $\mathcal{U}(0,1)$ is used to reject the new sample if $\alpha > A(\tilde{X}_{t-1}^{(j)}, \tilde{X}_t^{*(j)})$ and retain the previous sample ($\tilde{X}_t^{(j)} = \tilde{X}_{t-1}^{(j)}$). Now, the particles $\tilde{S}_t$ goes through the *importance sampling* module and then *diffusion* module to follow the particle filter approach. We denote this method as PI-MCPF for the rest of the chapter.

3.5 Experimental Details and Results

In this section, we give details about our implementation. We compare the proposed methods (PI-PF and PI-MCPF) with a baseline ICP based method on the primitive object interaction cases. We report our observations and demonstrate the methods on complex scenes. We use Blender v2.74 [112] binaries, along with its Python support and built-in implementation of Bullet [113] physics simulator. Prior to the experiment, a template scene is created in Blender with a camera, 3D object meshes and a supporting surface that acts as the table. We used real world objects with cuboid geometry for our experiments, whose object meshes are trivial to create in Blender using their real dimensions. For every experiment, the system is provided with the number of objects in the scene and their geometries in the form of meshes. We assume that an ideal recognition system provides this information without localizing the geometries in the scene. We used the default density value (1.0) in Blender for our experiments, which makes the object mass equal to its volume. All the object meshes in the scene are set as *active rigid bodies*, which means they react to collision and are subjected to gravitational forces. The supporting surface created is set to behave as a *passive rigid body*, which means it reacts to collisions but is not subjected to gravity (i.e. it interacts with objects but stays fixed in the scene). A Microsoft Kinect depth sensor mounted on top of the PR2 is externally calibrated with respect to the table using AR_Marker package *ar_track_alvar* from ROS providing extrinsic parameters. This calibration helps in creating a virtual supporting surface in Blender. After the template scene's blend file is created, at every iteration of the particle filter, the S_t set of scenes are loaded in parallel on multiple instances of Blender. In each instance, a particle X_t^j is loaded to set the pose of the object meshes and then physics rigid body simulation is triggered

to project each of the states from X_t^j to $\tilde{X}_t^j$. Blender rigid body simulation requires few critical parameters: we set up the friction coefficient to 0.75, rigid body sim frame_end at 500 (threshold to end the simulation), solver iterations at 60 and steps per second at 750. We found these parameters to be optimal for realistic physics simulation of the cuboid geometries considering its computation time. Depth images are rendered in HDR format to extract the exact metric information from the OpenGL renderer of Blender. We used 1444 particles for all our experiments. For primitive cases, PI-PF method was run for 150 iterations and PI-MCPF method was run for 70 iterations. For complex scenes, PI-PF method was run for 250 iterations and PI-MCPF method was run for 150 iterations.

In the below subsections, we discuss the implementation of baseline ICP method and compare its results with our proposed methods on the primitive cases considered in Section III. For the base clutter scenes, we created scenes which are difficult with insufficient depth data for traditional discriminative methods of object segmentation, object detection or pose estimation to perform robustly. The base clutter experiments involves two objects in touching, stacking and slant positions and also in complete occlusion. We experiment on 7 touching scenes, 7 stacking scenes, 7 slant cases and 7 complete occlusion scenes.

3.5.1 Iterative Closest Point method

Iterative Closest Point (ICP) [114] or its variants [115, 116, 117] are commonly sort after as the final step of pose estimation in the works that resulted from Amazon Pickup Challenge (APC) [118]. Hence we created a baseline with Iterative Closest Point (ICP) to estimate object poses in a scene. ICP takes in two point clouds namely the source cloud and the target cloud, and finds the transformation between them by iteratively by minimizing their point-to-point distance. This procedure requires the source and target to contain the same object to perform optimally. To provide this advantage to ICP based method, the 3D point cloud of each scene in the base clutter cases is processed in two stages: 1) the table background is subtracted by removing the largest plane in the scene using plane segmentation from PCL (Point Cloud Library) [119] resulting in a foreground point cloud of interest 2) each of the two objects are manually segmented from the foreground cloud resulting in two object point clouds (as the base clutter scene experiments contain only two objects). Point cloud of each object geometry is synthetically generated based on their dimensions and considered as source clouds for ICP matching. Each of these source clouds are matched with their respective target clouds segmented from the the scene. ICP matching is prone to be sensitive to the initialization of the source point cloud. Initial position (x, y, z) of the source clouds are generated randomly above the table level. The orientation (φ, θ, ψ) of these source clouds are set to the 3 principle components of their respective target clouds. For each scene, 50 randomly

Category	Error	ICP on Object Segments		PI-PF		PI-MCPF	
		Large Obj (mean ± var)	Small Obj (mean ± var)	Large Obj (mean ± var)	Small Obj (mean ± var)	Large Obj (mean ± var)	Small Obj (mean ± var)
Touching	Pos (cm)	9.58 ± 4.99	10.7 ± 3.68	**1.83** ± 0.18	**1.75** ± 0.11	2.10 ± 0.15	2.10 ± 0.50
	Roll (deg)	25.9 ± 4.51	62.0 ± 0.14	0.19 ± 0.05	0.30 ± 0.20	**0.17** ± 0.05	**0.23** ± 0.19
	Pitch (deg)	34.0 ± 2.71	38.8 ± 2.61	0.05 ± 0.00	**0.05** ± 0.01	**0.03** ± 0.00	**0.05** ± 0.00
	Yaw (deg)	28.8 ± 2.03	33.1 ± 8.45	**1.86** ± 3.06	**1.10** ± 0.58	2.30 ± 3.23	8.70 ± 3.06
Stacked	Pos (cm)	11.3 ± 1.83	13.0 ± 0.94	2.19 ± 0.60	**2.23** ± 0.20	**1.84** ± 0.99	2.67 ± 0.85
	Roll (deg)	32.2 ± 0.13	37.6 ± 0.54	0.53 ± 0.37	0.77 ± 1.13	**0.48** ± 0.57	**0.79** ± 1.77
	Pitch (deg)	37.1 ± 0.63	26.2 ± 2.20	**1.09** ± 3.81	1.54 ± 2.59	1.35 ± 5.45	**1.18** ± 3.19
	Yaw (deg)	57.5 ± 3.04	38.4 ± 2.59	**4.71** ± 6.74	**6.05** ± 5.86	5.50 ± 8.43	6.63 ± 8.32
Slant	Pos (cm)	10.4 ± 5.23	14.3 ± 0.72	**3.09** ± 5.51	4.38 ± 11.4	4.42 ± 5.90	**4.33** ± 6.82
	Roll (deg)	36.9 ± 8.97	39.3 ± 2.50	14.5 ± 86.5	0.38 ± 0.10	**0.54** ± 1.02	**0.33** ± 0.10
	Pitch (deg)	38.8 ± 0.29	33.4 ± 2.94	**1.58** ± 2.97	31.5 ± 23.3	5.96 ± 69.3	**19.4** ± 74.3
	Yaw (deg)	19.9 ± 2.78	27.6 ± 1.93	10.5 ± 84.3	**30.7** ± 42.4	**10.3** ± 19.9	36.5 ± 31.6
Occluded	Pos (cm)	26.7 ± 2.33	NA	**2.83** ± 1.47	**4.23** ± 5.65	3.23 ± 2.38	4.28 ± 5.63
	Roll (deg)	13.8 ± 3.37	NA	**20.0** ± 71.1	**29.9** ± 43.6	**20.0** ± 72.8	44.9 ± 44.8
	Pitch (deg)	8.47 ± 1.10	NA	**0.05** ± 0.00	**30.0** ± 85.3	**0.05** ± 0.00	**30.0** ± 87.5
	Yaw (deg)	27.5 ± 3.40	NA	**15.0** ± 53.6	40.0 ± 40.0	16.1 ± 18.1	**33.9** ± 49.8

Table 3.1: Object pose estimation errors are reported here with respect to the ground truth poses. Ground truth is generated by manually matching the object geometries to the observed point cloud using the Blender user interface. In all the experimental categories (touching, stacked, slant and occluded), physical informed estimators PI-PF, PI-MCPF perform better than the ICP method. The variance of the physics informed methods are higher in the slant cases as the simulations result in different plausible slant pose every time. In the occluded category of experiments, the ICP method has NA entries as the method is not applicable when no sensor data is available.

initialized source clouds of the objects are used to perform the ICP matching.

3.5.2 Base Clutter Scene Results

In the touching cases, two objects are placed in different orientations on the table, touching each other as shown in Fig. 3.3. We show the cases where objects are in contact on their edges or their faces. It is observed that the estimates of these cases using PI-PF and PI-MCPF methods are close to the ground truth with average errors in position and angles as shown in Table 3.1. ICP on the object segments fail with large pose errors as they are not physically informed about their object boundaries leading to inter-penetrations.

In the stacking cases, two objects are placed in different orientations on table, with one object placed on top of the other object. This other object is supported by the table as shown in Fig. 3.4. Note, that we used only small objects to be on top of the larger object, because the converse structure creates complete occlusion, which is discussed in the following set of experiments. It is observed that, in order to generate stacking scenes using physics projection, the diffusion of the resampled $\tilde{S}_t$ states should accommodate elevation of objects randomly. This diffusion creates S_t. We observed that the estimated scenes using PI-PF and PI-MCPF methods are close to the ground truth with average errors in position and angles as shown in Table 3.1. ICP based approach fails to perform as the objects are not enforced to stack based on their poses and hence could result in

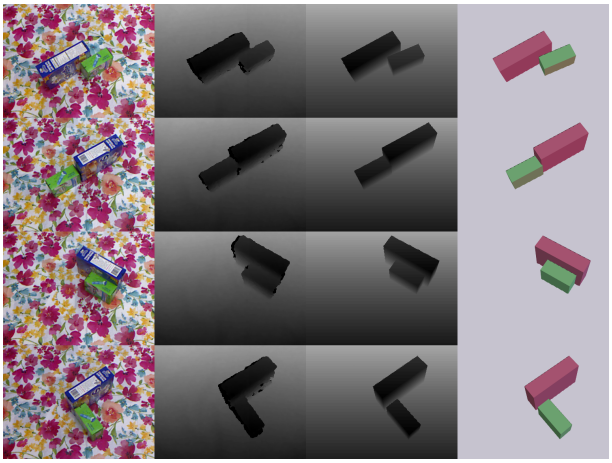


Figure 3.3: Results for objects touching experiment: From left Original Scene, Observed depth image, Estimated most likely scene as a depth image, Blender camera view of the estimated scene

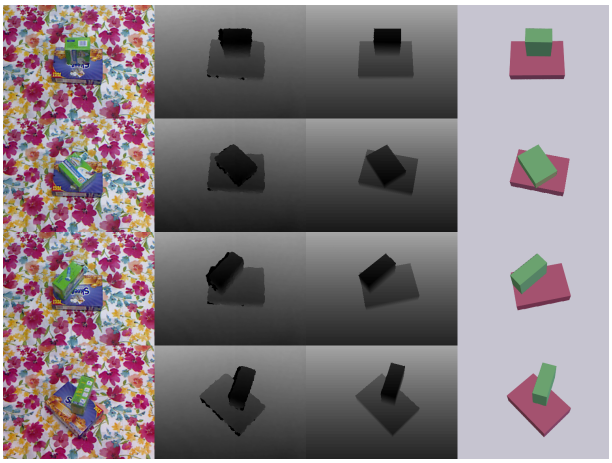


Figure 3.4: Results for objects stacking experiment: From left Original Scene, Observed depth image, Estimated most likely scene, Blender camera view of the estimated scene

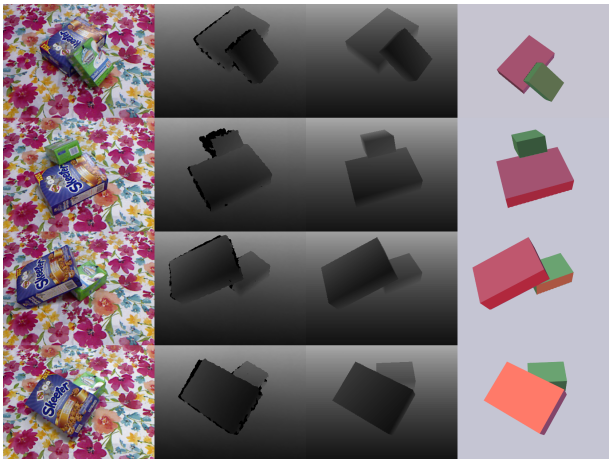


Figure 3.5: Results for objects slanted experiment: From left Original Scene, Observed depth image, Estimated most likely scene, Blender camera view of the estimated scene

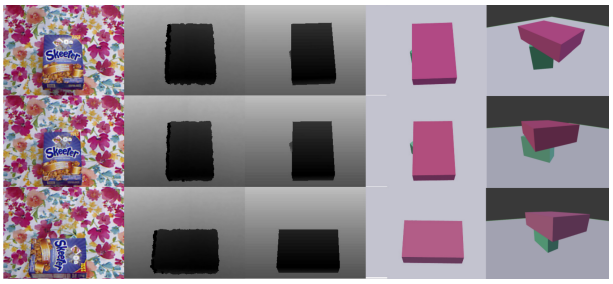


Figure 3.6: Results for objects occluded experiment: From left Original Scene, Observed depth image, Estimated most likely scene, Blender camera view of the estimated scene with an additional view to show how the occluded object‘s pose is estimated by our method

floating objects.

In the slant cases, two objects are placed in different orientations on table such that one object is on the table, supporting the other object, which is in a slant pose as shown in Fig. 3.5. To generate slant scenarios, the rigid body simulation in Blender requires care in setting up the parameters as mentioned above. If physics projection cannot produce these slant cases, the experiments will not converge to the observed scene. As it can be seen in Fig. 3.5, even in the cases where the bottom object is occluded by the top slant object, its pose in the estimated depth image matches the observation. More importantly, we find that estimated state is physically plausible. We observed that slant cases are difficult, and estimates from both PI-PF and PI-MCPF methods are not as close to the ground truth as in touching and stacked cases. The average angular error is high for the small object, which is occluded in most of the cases and very hard to be estimated. On the other hand, the larger object which, even on having an advantage of being highly visible requires a trade off in matching the observation and also maintaining physical plausibility. ICP fails in slant cases too as it is not informed about both the object boundaries as well as gravitational force to support itself in a slant position.

In the occlusion cases, as shown in Fig. 3.6, the small object is completely occluded by the larger object in the observation. Our proposed methods PI-PF and PI-MCPF robustly handles these cases and estimates the pose of the larger object with average position errors shown in Table 3.1. However these methods have higher position errors for the smaller object that is not visible to the sensor. It should be noted that the ground truth for all these scenes were generated using visual inspection and matching of the object geometries to the observed point cloud. Because the small object was not seen in the point cloud, the ground truth was generated to just make sure physical plausibility of the scene. The last column in Fig. 3.6 shows the view of the estimated scene from a different viewpoint, to see the estimated pose of the occluded small object. In complete occlusion, we also had cases where the larger object was slanted on the small object, occluding the small object. Hence there is a high error in the *Roll* of the larger object similar to that of the slant cases in both PI-PF and PI-MCPF methods. ICP based method does not have the target cloud for the small

object, and, thus there is no way to estimate the pose of that object.

The ICP based method purely relies on 3D data association. It is observed to fail consistently on all categories. It should also be noted that ICP will perform much worse if the 3D scene is not preprocessed. Overall the PI-PF and PI-MCPF methods perform comprehensively on these difficult primitive setups and help us develop an understanding of using physical plausibility in the estimation process of more complex scenes discussed in next section.

3.5.3 Cluttered Scene Results

We have performed experiments on three and four objects cases, that combined the base cases discussed earlier. With inclusion of additional objects, the state space for search explodes and it takes lot of iterations to converge to the ground truth. For experimental purpose the time complexity is avoided with constrains on the object poses. Poses of the objects are limited to $\{x_i, y_i, z_i, \varphi_i\}$ (i.e. φ_i is the yaw angle of an object to determine its rotation on the surface plane which is aligned to XY plane) dimensions in the initialization and updates. However physics projections at each iteration results in real valued numbers on all the $6xN$ dimensions of the scene.

In Fig. 3.7 we show experiments with four objects in the scene with results from PI-PF. It can be seen that the experimental set up has the combinations of the primitive cases discussed earlier. These scenes have a lot of occlusions with respect to the sensor viewpoint. The scenes are estimated using PI-PF and PI-MCPF, and are close to the ground truth poses, except for the objects that are occluded. However if a continuous perception is performed, our estimation along with the distribution over the state space will act as a prior knowledge over time. We performed sequence of object manipulations on the estimated scenes using PR2 robot, whose gripper has a small tolerance to the error in estimation. Precision to which the pose estimation is performed in PI-PF and PI-MCPF methods are good enough to let the robot perform successful manipulation. A couple of scenes are shown in the video submission with robotic manipulation on the estimated poses. We observed that the accuracy of the PI-PF and PI-MCPF are close to each other in all the experiments performed, but the number of iterations taken by PI-PF is higher compared to PI-MCPF as shown in Table 3.2.

3.6 Summary

In this chapter, we proposed a generative, probabilistic scene estimation using Newtonian physical simulation for physically plausible scene estimation to enable robotic manipulation in clutter. Our method estimates cluttered scenes as a collection of object poses to generate and match observation. We discuss primitive cases causing observation uncertainty due to object interactions like

Conditions	Maximum iterations allowed	Average iterations for PI-PF	Maximum iterations allowed	Average iterations for PI-MCPF
Touching	150	85.26	70	30.42
Stacking	150	90.84	70	53.55
Slant	150	143.7	70	70.00
Occlusion	150	70.50	70	46.98
4 Objects	250	224.6	150	142.1

Table 3.2: Shows the average number of iterations each of methods, took to converge. Maximum iterations are the number of iterations each method is allowed to run. We consider the experiment to have converged if the change in the pose estimate of the most likely particle is less than 1 cm in position and less than 3 degrees in the angles.

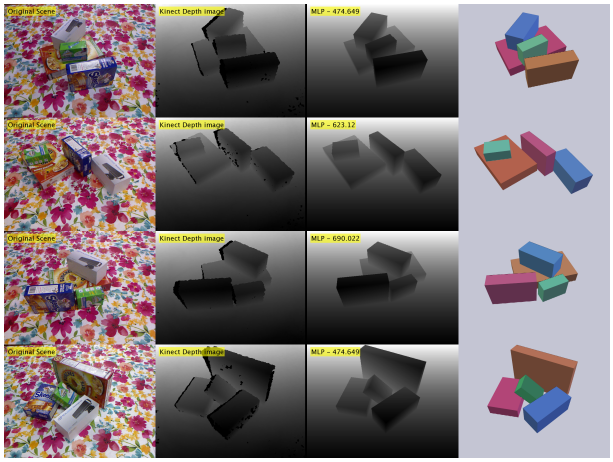


Figure 3.7: Results for complex experiment with 4 objects: From left to right, Original Scene, Observed depth image, Estimated most likely scene, Blender camera view of the estimated scene

touching, stacking and slant support poses. We present cases where physical plausibility is at most essential in robotic perception and show results using our framework on some difficult cases of clutter settings. We explored variants of our approach and report the results with observations on each case.

Subsequently and independently to our published work [15], Mitash et al. [120] developed object pose estimation system that improves over the use of physical context. In contrast to our work that considers scene as a whole while generatively producing physically plausble states, the work [120] searches over physically consistent scene configuration using Monte Carlo tree search driven by discriminative object detection and heuristics. Together these works substantiate the need for considering physical context in scene estimation towards manipulation applications. However, the limitations of our approach are: a) exponential growth in the state space with the number of objects and b) physical simulators are limited when it come to simulating arbitrarily shaped objects that has functional parts. We address these two limitations in Chapter 4.

CHAPTER 4

Efficient Belief Propagation for Pose Estimation of Articulated Objects

4.1 Introduction

Going back to the motivating scenario shown in the Figure. 1.2 of Chapter 1, robots working in human environments often encounter a wide range of articulated objects, such as tools, cabinets, and other kinematically jointed objects. For example, the tool chest with four drawers in Figure. 1.2 functions as a storage container for tools. A robot would need to perform open and close actions on the drawers and store the tools from the pile towards accomplishing the task of setting up the workbench.

In addition to the rigid bodied objects, the ability to perceive articulated pose under partial observations due to self and environmental occlusions makes the inference problem multimodal. Further, the inference becomes a high-dimensional problem when the number of object parts grows in number, which is one of the limitations of Chapter 3.

Pose estimation methods have been proposed that take a generative approach to the problem [13, 1], including the work from the Chapter 3 [15]. These methods aim to explain a scene as a collection of object/parts poses, using a particle filter formulation to iteratively maintain belief over possible states in the form of particles. Though these approaches hold the power of modeling the world generatively, they have an inherent drawback of being slow with the increase in the number of rigid bodies. In this chapter, we focus on overcoming this drawback by factoring the state as individual object and object parts constrained by their physical support relations to create an efficient inference framework for pose estimation. In this chapter, we specifically focus on articulated objects to draw comparisons with other factored methods in the literature. However, the approach is still applicable to both objects with and without articulations.

Generative methods exploiting articulation constraints are widely used in human pose estimation problems [14, 64, 98] where human body parts have constrained articulation. We take a similar approach and factor the problem using a Markov Random Field (MRF) formulation where each

hidden node in the probabilistic graphical model represents an observed object-part's pose (continuous variable), each observed node has the information about the object-part from observation and edges of the graph denote the articulation constraints between the parts. Inference on the graph is performed using a message passing algorithm that shares the information between the parts' pose variables, to produce their pose beliefs which collectively gives the state of the articulated object.

Existing message passing approaches [80, 79] represent message as a mixture of Gaussian components and provide Gibbs sampling-based techniques to approximate message product and update operations. Their message representation and product techniques limit the number of samples used in the inference and is not applicable to our application domain. In this chapter, we provide a more efficient "pull" Message Passing algorithm for Nonparametric Belief Propagation (PMPNBP) [16, 17]. The key idea of pull message updating is to evaluate samples taken from the belief of the receiving node with respect to the densities informing the sending node. The mixture product approximation can then be performed individually per sample, and then later normalized into a distribution. This pull updating avoids the computational pitfalls of push updating of message distributions used in [80, 79]. We demonstrate the accuracy and efficiency of inference by PMPNBP with respect to PAMPAS [80], a pioneering method for NBP. These results focus on an experiment for finding an articulated 2D pattern, reconstructed from the description of PAMPAS. These results indicate that PMPNBP enables both faster convergence to an appropriate inference and greater scaling of message mixture components for improved accuracy.

For our robot experiments, our system takes in a 3D point cloud as the sensor data and object geometry models in the form of a URDF (Unified Robot Description Format) as input and outputs belief samples in continuous pose domain. We use these belief samples to compute a maximum likely estimate to let the robot act on the object. We evaluate the performance of the system by quantifying over an articulated object on compelling scenes. Contributions of this chapter include: a) proposal of an efficient belief propagation algorithm to estimate articulated object poses, b) discussion and comparisons with PAMPAS [80] on the 2D baseline, c) discussion and comparisons with traditional particle filter as the baseline, d) a belief representation from perception to inform a task planner. A simple task is demonstrated to show how the belief propagation informs a task planner to choose an information gain action and overcome uncertainty in the perceptual estimation.

4.2 Related Work

Indoor environments have a variety of objects that has functional properties associate with their parts. Physical constraints on these kinds of objects are in the form of articulation constraints imposed during their design. For example, the drawers of a cabinet are designed to have a transla-

tional motion to open and close the storage space. Similarly, a dishwasher's door opens with joint rotational constraints. There are many objects in the indoor environments robot has to deal with in order to perform household tasks.

Existing methods in the literature have set out to address the challenge of manipulating articulated objects by robots in complex human environments. Particular focus has been placed on addressing the task of estimating novel articulated objects' kinematic models by a robot through interactive perception. Hausman et al. [73] propose a particle filtering approach to estimate articulation models and plan actions to reduce model uncertainty. In [74], Martin et al. suggest an online interactive perception technique for estimating kinematic models by incorporating low-level point tracking and mid-level rigid body tracking with a high-level kinematic model estimation over time. Sturm et al. [75, 76] addressed the task of estimating articulation models in a probabilistic fashion by a human demonstration of manipulation examples. Katz et al. [121] also pair a RANSAC-based plane fitting algorithm and iterative rectangle search algorithm to perform marker-less pose estimation; however, they do not leverage their previously learned kinematic models.

All of these approaches discover the articulated object's kinematic model by alternating between action and sensing and are important methods to enable a robot is to interact with novel articulated objects reliably. In this chapter, we assume that such kinematic models, once learned for an object, can be reused to localize their articulated pose under real-world ambiguous observations. The method proposed here could complement the existing body of work towards task completion in the unstructured human environment with articulated objects.

Probabilistic graphical model representations such as Markov random field (MRF) are widely used in computer vision problems where the variables take discrete labels such as foreground or background. Many algorithms have been proposed to compute the joint probability of the graphical model. Belief propagation algorithms are guaranteed to converge on tree-structured graphs. For graph structures with loops, Loopy Belief Propagation (LBP) [122] is empirically proven to perform well for discrete variables. The problem becomes non-trivial when the variables take continuous values. Sudderth et al. (NBP) [79] and Particle Message Passing (PAMPAS) by Isard et al. [80] provide sampling approaches to perform belief propagation with continuous variables. Both of these approaches approximate a continuous function as a mixture of weighted Gaussians and use local Gibbs sampling to approximate the product of mixtures. NBP has been effectively used in applications such as human pose estimation [14] and hand tracking [64] by modeling the graph as a tree-structured particle network. Scene understanding problems where a scene is composed of household objects with articulations demands a large number of samples in the representation to handle the high-dimensional multimodal state space. A Pull Message Passing algorithm for Nonparametric Belief Propagation (PMPNBP) proposed in this chapter produces promising results to handle such demands.

Some recent works address the computational efficiency of Nonparametric Belief Propagation. Similar in spirit to our PMPNBP, Ihler et al. [88] describe a conceptual theory of particle belief propagation, where a target node's samples are used to generate a message going from source to target. This work emphasizes the advantages of using a large number of particles to represent incoming messages and theoretical analysis. This work uses an expensive iterative Markov Chain Monte Carlo sampling step, mimicking the Gibbs sampling step in NBP [80, 79]. PMPNBP can avoid this cost through a resampling step.

Kernel based methods have been proposed to improve the efficiency of NBP. Song et al. [123] propose a kernel belief propagation method. In this work, messages are represented as functions in a Reproducing Kernel Hilbert space (RKHS), and message updates are linear operations in RKHS. Results presented in this work claim to be more accurate and faster than NBP with Gibbs sampling [80, 79] and particle belief propagation [88] over applications such as image denoising, depth prediction, and angle prediction in protein folding problem. We consider comparisons with kernel-based approximators as a direction for future work.

Model-based generative methods [12, 57, 6] are increasingly being used to solve scene estimation problems where heuristics from discriminative approaches [124, 125] are used to infer object poses. These approaches do not account for object-object interactions or articulations and rely significantly on the effectiveness of recognition. Our framework does not rely on any prior detections but can benefit from them while inherently handling noisy priors [79, 80, 126]. Chua et al. [127] proposed a scene grammar representation and belief propagation over factor graphs, whose objective similar to ours for generating scenes with multiple-objects satisfying the scene grammars. This approach is similar to ours; however, we specifically deal with 3D observations and continuous variables.

4.3 Method

4.3.1 Nonparametric Belief Propagation

Let $G = (V, E)$ be an undirected graph with nodes V and edges E. The nodes in V are each random variables that have dependencies with each other in the graph G through edges E. If G is a Markov Random Field (MRF), then it has two types of variables X and Y, denoting the collection of hidden and observed variables, respectively. Each variable is considered to take assignments of continuous-valued vectors. The joint probability of the graph G, considering only second order cliques, is given as

$$p(X, Y) = \frac{1}{Z} \prod_{(s,t)\in E} \psi_{s,t}(X_s, X_t) \prod_{s\in V} \phi_s(X_s, Y_s) \tag{4.1}$$

where $\psi_{s,t}(X_s, X_t)$ is the pairwise potential between nodes $X_s \in \mathbb{R}^d$ and $X_t \in \mathbb{R}^b$ [1], $\phi_s(X_s, Y_s)$ is the unary potential between the hidden node X_s and observed node $Y_s \in \mathbb{R}^q$, and Z is a normalizing factor. The problem is to infer belief over possible states assigned to the hidden variables X such that the joint probability is maximized. This inference is generally performed by passing messages between hidden variables X until convergence of their belief distributions over several iterations.

A message is denoted as $m_{t\to s}$ directed from node t to node s if there is an edge between the nodes in the graph G. The message represents the distribution of what node t thinks node s should take in terms of the hidden variable X_s. Typically, if X_s is in the continuous domain, then $m_{t\to s}(X_s)$ is represented as a Gaussian mixture to approximate the real distribution:

$$m_{t\to s}(X_s) = \sum_{i=1}^{M} w_{ts}^{(i)} \mathcal{N}(X_s; \mu_{ts}^{(i)}, \Lambda_{ts}^{(i)}) \tag{4.2}$$

where $\sum_{i=1}^{M} w_{ts}^{(i)} = 1$, M is the number of Gaussian components, $w_{ts}^{(i)}$ is the weight associated with the i^{th} component, $\mu_{ts}^{(i)}$ and $\Lambda_{ts}^{(i)}$ are the mean and covariance of the i^{th} component, respectively. We use the terms components, particles and samples interchangeably in this chapter. Hence, a message can be expressed as M triplets:

$$m_{t\to s} = \{(w_{ts}^{(i)}, \mu_{ts}^{(i)}, \Lambda_{ts}^{(i)}) : 1 \le i \le M\} \tag{4.3}$$

Assuming the graph has tree or loopy structure, computing these message updates is nontrivial computationally. A message update in a continuous domain at an iteration n from a node $t \to s$ is given by

$$m_{t\to s}^{n}(X_s) \leftarrow \int_{X_t \in \mathbb{R}^b} \left(\psi_{st}(X_s, X_t) \phi_t(X_t, Y_t) \prod_{u \in \rho(t)\setminus s} m_{u\to t}^{n-1}(X_t) \right) dX_t \tag{4.4}$$

where $\rho(t)$ is a set of neighbor nodes of t. The marginal belief over each hidden node at iteration n is given by

$$\begin{aligned} bel_s^n(X_s) &\propto \phi_s(X_s, Y_s) \prod_{t \in \rho(s)} m_{t\to s}^{n}(X_s) \\ bel_s^n &= \{(w_s^{(i)}, \mu_s^{(i)}, \Lambda_s^{(i)}) : 1 \le i \le T\} \end{aligned} \tag{4.5}$$

where T is the number of components used to represent the belief. NBP [79] provides a Gibbs sampling approach to compute an approximation of the product $\prod_{u \in \rho(t)\setminus s} m_{u\to t}^{n-1}(X_t)$. Assuming that $\phi_t(X_t, Y_t)$ is pointwise computable, a "pre-message" [88] is defined as

$$M_{t\to s}^{n-1}(X_t) = \phi_t(X_t, Y_t) \prod_{u \in \rho(t)\setminus s} m_{u\to t}^{n-1}(X_t) \tag{4.6}$$

[1] Note, dimensionality remains the same, $d = b$, in the case of estimating 6 degree-of-freedom object pose

Algorithm - Message update

Given input messages $m^{n-1}_{u\to t}(X_t) = \{(\mu^{(i)}_{ut}, w^{(i)}_{ut})\}^M_{i=1}$ for each $u \in \rho(t) \setminus s$, and methods to compute functions $\psi_{ts}(X_t, X_s)$ and $\phi_t(X_t, Y_t)$ point-wise, the algorithm computes $m^n_{t\to s}(X_s) = \{(\mu^{(i)}_{ts}, w^{(i)}_{ts})\}^M_{i=1}$

1. Draw M independent samples $\{\mu^{(i)}_{ts}\}^M_{i=1}$ from $bel^{n-1}_s(X_s)$.
   (a) If $n = 1$ the $bel^0_s(X_s)$ is a uniform distribution or informed by a prior distribution.
   (b) If $n > 1$ the $bel^{n-1}_s(X_s)$ is a belief computed at $(n-1)^{th}$ iteration using importance sampling.

2 For each $\{\mu^{(i)}_{ts}\}^M_{i=1}$, compute $w^{(i)}_{ts}$

   a Sample $\hat{X}^{(i)}_t \sim \psi_{ts}(X_t, X_s = \mu^{(i)}_{ts})$

   b Unary weight $w^{(i)}_{unary}$ is computed using $\phi_t(X_t = \hat{X}^{(i)}_t, Y_t)$.

   c Neighboring weight $w^{(i)}_{neigh}$ is computed using $m^{n-1}_{u\to t}$.

      (i) For each $u \in \rho(t) \setminus s$ compute $W^{(i)}_u = \sum^M_{j=1} w^{(j)}_{ut} w^{(ij)}_u$ where $w^{(ij)}_u = \psi_{ts}(X_s = \mu^{(i)}_{ts}, X_t = \mu^{(j)}_{ut})$.

      (ii) Each neighboring weight is computed by $w^{(i)}_{neigh} = \prod_{u\in\rho(t)\setminus s} W^{(i)}_u$

   d The final weights are computed as $w^{(i)}_{ts} = w^{(i)}_{neigh} \times w^{(i)}_{unary}$.

3 The weights $\{w^{(i)}_{ts}\}^M_{i=1}$ are associated with the samples $\{\mu^{(i)}_{ts}\}^M_{i=1}$ to represent $m^n_{t\to s}(X_s)$.

which can be computed in the Gibbs sampling procedure. This reduces Equation 4.4 to

$$m^n_{t\to s}(X_s) \leftarrow \int_{X_t\in\mathbb{R}^b} \Big(\psi_{st}(X_s, X_t) M^{n-1}_{t\to s}(X_t)\Big) dX_t \tag{4.7}$$

The pairwise term $\psi_{st}(X_s, X_t)$ can be approximated as the marginal influence function $\zeta(X_t)$ to make the right side of Equation 4.7 independent of X_s. The marginal influence function provides the influence of X_s for sampling X_t. However, this function can be ignored if the pairwise potential function is based on the distance between the variables. This assumption makes Equation 4.7 avoid the step of integration and sample $\hat{X}^{(i)}_t$ from the "pre-message" followed by a pairwise sampling where $\psi_{st}(X_s, X_t)$ is acting as $\psi_{st}(X_s|X_t = \hat{X}^{(i)}_t)$ to get a sample $\hat{X}^{(i)}_s$. To represent message $m^n_{t\to s}(X_s)$, the M samples $\{\hat{X}_s\}^M_{i=1}$ are considered as $\{\mu_{ts}\}^M_{i=1}$. $\{\Lambda_{ts}\}^M_{i=1}$ are computed using Kernel Density Estimation methods. PAMPAS [80] has a slightly different notation and methods to compute the samples.

The Gibbs sampling procedure in itself is an iterative procedure and hence makes the compu-

Algorithm - Belief update

Given incoming messages $m^n_{t\to s}(X_t) = \{(w^{(i)}_{ts}, \mu^{(i)}_{ts})\}^M_{i=1}$ for each $t \in \rho(s)$, and methods to compute functions $\phi_s(x_s, y_s)$ point-wise, the algorithm computes $bel^n_s(X_s) \propto \phi_s(X_s, Y_s) \prod_{t\in\rho(s)} m^n_{t\to s}(X_s) = \{(w^{(i)}_s, \mu^{(i)}_s)\}^T_{i=1}$

1 For each $t \in \rho(s)$

   a Update weights $w^{(i)}_{ts} = w^{(i)}_{ts} \times \phi(X_s = \mu^{(i)}_{ts}, Y_s)$.

   b Normalize the weights such that $\sum^M_{i=1} w^{(i)}_{ts} = 1$.

2 Combine all the incoming messages to form a single set of samples and their weights $\{(w^{(i)}_s, \mu^{(i)}_s)\}^T_{i=1}$, where T is the sum of all the incoming number of samples.

3 Normalize the weights such that $\sum^T_{i=1} w^{(i)}_s = 1$.

4 Perform a resampling step to sample new set $\{\mu^{(i)}_s\}^T_{i=1}$ that represent the marginal belief of X_s.

tation of the "pre-message" (as the Foundation function described for PAMPAS) expensive as M increases. In the next section, we provide our proposed message representation followed by the algorithm to compute $m^n_{t\to s}(X_s)$ at iteration n.

4.3.2 Pull Message Passing for Nonparametric Belief Propagation

Given the overview of Nonparametric Belief Propagation above in Section 4.3.1, we now describe our "pull" message passing algorithm. We represent message as a set of pairs instead of triplets in Equation 4.3 which is

$$m_{t\to s} = \{(w^{(i)}_{ts}, \mu^{(i)}_{ts}) : 1 \le i \le M\} \tag{4.8}$$

Similarly, the marginal belief is summarized as a sample set

$$bel^n_s(X_s) = \{\mu^{(i)}_s : 1 \le i \le T\} \tag{4.9}$$

where T is the number of samples representing the marginal belief. We assume that there is a marginal belief over X_s as $bel^{n-1}_s(X_s)$ from the previous iteration. To compute the $m^n_{t\to s}(X_s)$, at iteration n, we initially sample $\{\mu^{(i)}_{ts}\}^M_{i=1}$ from the belief $bel^{n-1}_s(X_s)$. Pass these samples over to the neighboring nodes $\rho(t) \setminus s$ and compute the weights $\{w^{(i)}_{ts}\}^M_{i=1}$. This step is described in Algorithm - Message update. The computation of $bel^n_s(X_s)$ is described in Algorithm - Belief update. The key difference between the "push" approach of the earlier methods (NBP and PAMPAS) [79, 80]

and our "pull" approach is the message $m_{t\to s}$ generation. In the "push" approach, the incoming messages to t determines the outgoing message $t \to s$. Whereas, in the "pull" approach, samples representing s are drawn from its belief bel_s from previous iteration and weighted by the incoming messages to t. This weighting strategy is computationally efficient. Additionally, the product of incoming messages to compute bel_s is approximated by a resampling step as described in Algorithm - Belief update.

4.4 Experimental Details and Results

4.4.1 Comparison of Message Passing Algorithms on Articulated Pattern

We compare our proposed PMPNBP method with PAMPAS [80] on their 2D illustratory example (Figure 4.1). The pattern has circle node with state variable $X_1 = (x_1, y_1, r_1)$ denoting its position in the 2D image and the radius of the circle. This circle node has four arms with two links each. These links are nodes in the graph with state variables $X_i(x_i, y_i, \alpha_i, w_i, h_i)$. The links connected to the circle are indexed as $2 \leq i \leq 5$ with their connected outer links as $j = i + 4$. In the recreation of this illustratory example, we define the unary potential as

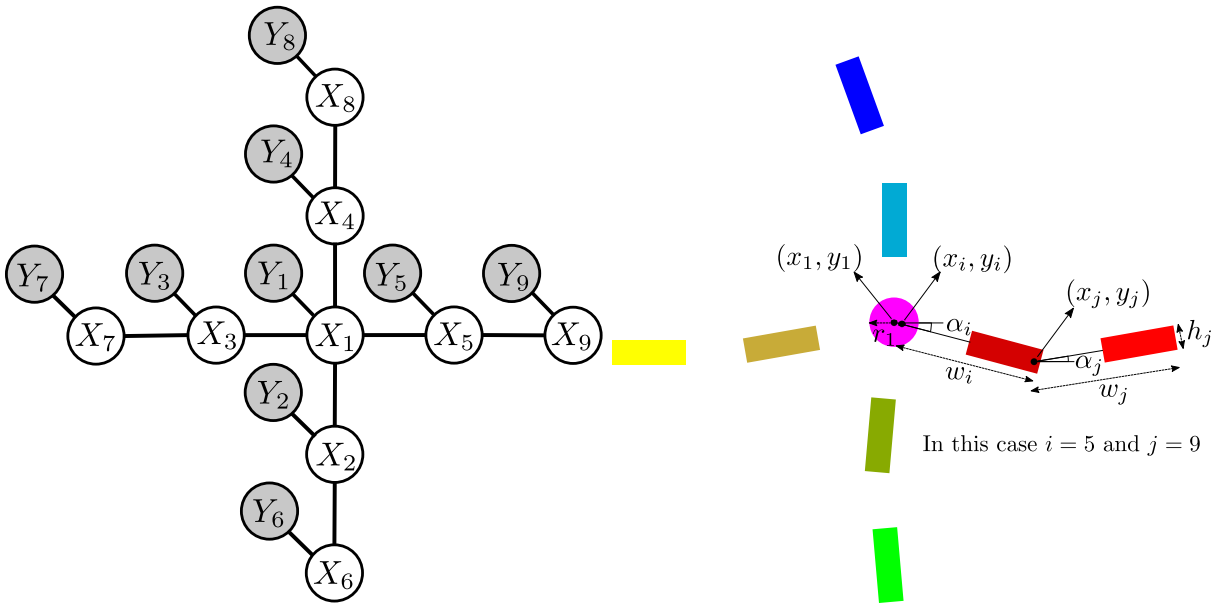
$$\phi(X_s, Y_s) = \begin{cases} 1 - \frac{|I_{sub}(\{(x_s,y_s)\}_{p=1}^{P}) - T(\{(x_s,y_s)\}_{p=1}^{P})|}{\max(P,Q)} & X_s \in \text{circle} \\ 1 - \frac{|I_{sub}(\{(x_s,y_s)\}_{p=1}^{P}) - T(\{(x_s,y_s,\alpha_s,w_s,h_s)\}_{p=1}^{P})|}{\max(P,Q)} & X_s \in \text{links} \end{cases} \tag{4.10}$$

where I_{sub} is the patch of image centered at (x_s, y_s) with the same size as the template image T rendered with state of the nodes (circle/links). P and Q are the number of white/observed pixel locations $\{(x, y)\}$ in I_{sub} and T respectively. Figure 4.2 illustrates the computation of the unary potential for nodes X_1, X_2, X_3 visually.

The pairwise sampling is done similar to the original description in PAMPAS [80]. The procedure to generate samples is described in Appendix A. Figure 4.3 visually illustrates the pairwise sampling for nodes X_1, X_5, X_9. With the unary potential and pairwise sampling, we perform inference and report their convergence over iterations in the next section.

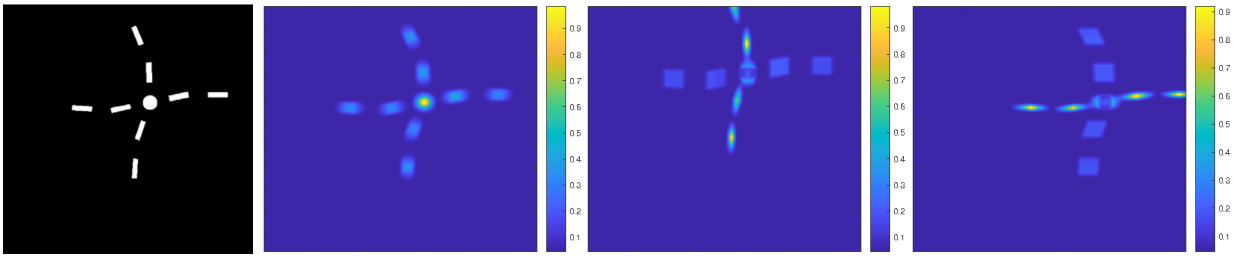
Our implementation of PAMPAS and PMPNBP is in Matlab on a Ubuntu 14.04 Linux machine. A CPU with Core i7 6700HQ - 16 GB RAM is used for all the experiments. Implementation does not involve any type of parallelization to avoid bias in comparisons.

We show the convergence of the PMPNBP qualitatively in Figure 4.4 and Figure 4.5. The pattern referred in Figure 4.1 is placed in a clutter made of 12 circles and 100 rectangles. There are 16 messages, i.e., 4 from circle to inner links, 4 from inner links to circle, 4 from inner links to outer links and 4 from outer to inner links. The initialization of the messages is done with



(a) Graphical model (b) Geometrical structure

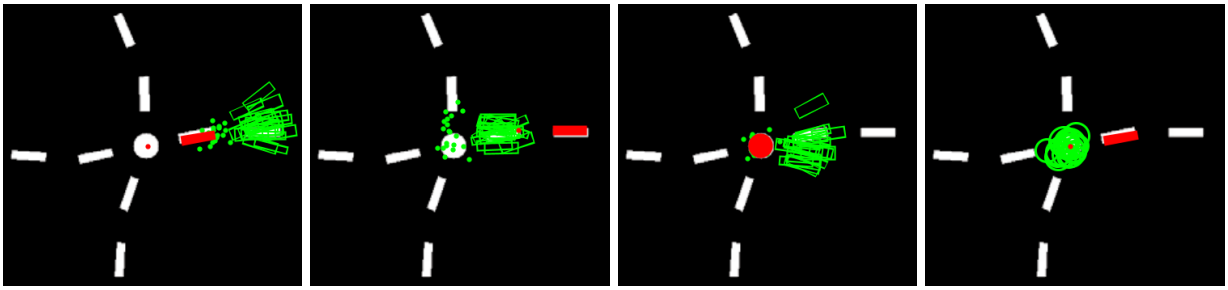
Figure 4.1: 2D articulation pattern and its graphical model: The pattern used for the experiments has 9 nodes with one circle at the center and four arms with two links each. This forms the graphical model shown in (a), where hidden nodes X_s are connected to their neighbors and informed by observed nodes Y_s. Geometrically, the circle and links are defined by their location (x_s, y_s), orientation and dimensions as shown in (b). Color coding here is used to distinguish the links for the qualitative results in the chapter.



(a) Image observation (b) $\phi(X_1, Y_1)$ (c) $\phi(X_2, Y_2)$ (d) $\phi(X_3, Y_3)$

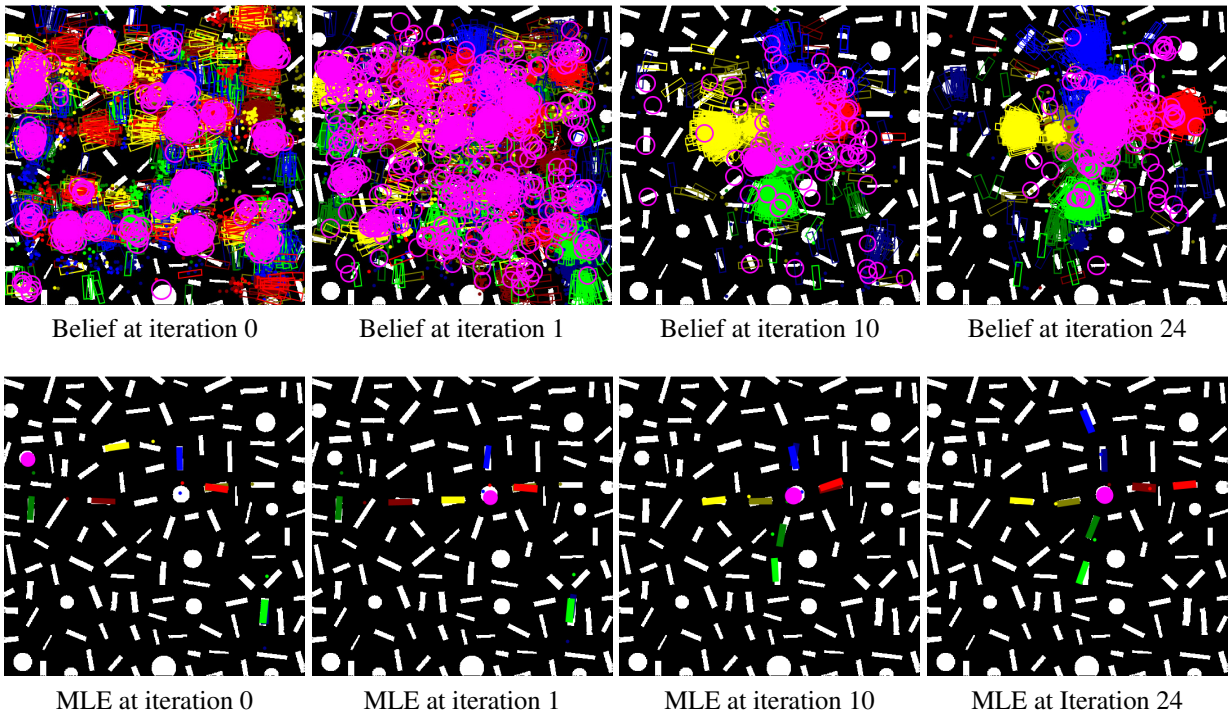
Figure 4.2: Illustration of unary potential for 2D articulation pattern: a) Shows the actual pattern used in the experiments of the chapter. b-c) shows the unary potential $\phi(X_s, Y_s)$ for $s = \{1, 2, 3\}$ (circle, vertical rectangular link and horizontal rectangular link respectively) with (x_s, y_s) taking all the pixels in the image (a). For ease of understanding, the orientation of the nodes in this illustration are set to $\alpha_1 = 0$, $\alpha_2 = \pi/2$ and $\alpha_3 = \pi$.

$M = 75$ particles at (x, y) locations of the image where $\phi_s > 0.4$. This is assumed to be the coarse feature detection of the circle and rectangles in the image replicating the initialization in [80]. In the future iterations, the message $m_{t \to s}$ has 50% of the samples uniformly sampled in the image to keep exploring, while the other 50% of the samples are sampled from the marginal belief bel_s. As it can be seen in Figure 4.4, the initialization (Belief at Iteration 0) is distributed across the image. At iteration 1, the message passing starts to influence the belief of the nodes and at iteration 10, they form the spatial arrangement satisfying their geometrical structure. At iteration 24, the most



(a) $\psi(X_9|X_5 = \mu_5)$ (b) $\psi(X_5|X_9 = \mu_9)$ (c) $\psi(X_5|X_1 = \mu_1)$ (d) $\psi(X_1|X_5 = \mu_5)$

Figure 4.3: Illustration of pairwise sampling for 2D articulation pattern: This figure shows sampling the neighbors based on a given current node sample. For illustration we show the relation between nodes X_1, X_5 and X_9. Each sub-figure shows 20 samples (green color) drawn given its neighboring node (red color) at its ground truth location, constrained by their geometrical relationship.



Belief at iteration 0 Belief at iteration 1 Belief at iteration 10 Belief at iteration 24

MLE at iteration 0 MLE at iteration 1 MLE at iteration 10 MLE at Iteration 24

Figure 4.4: Convergence in 2D articulation pose under clutter: PMPNBP results with circle node observed. Each message contains 200 particles initialized randomly at locations where their $\phi_s > 0.4$. The top row shows the belief samples bel_s for each of the nodes and the bottom row shows their Maximum Likely Estimate (MLE). MLE at iteration 24 has all the links and circle converged to their ground truth states (Best viewed in color).

likely estimate of all the links and circle are close to the pose of the ground truth pattern.

The second example in Figure 4.5 has no circle in the center of the pattern, demonstrating an occlusion scenario. This scenario demonstrates that the proposed algorithm retains the power of the probabilistic modeling of the belief propagation approach. The initialization is done similar to the first example, where there were no samples near the "occluded" circle. The convergence is

similar to the first example but takes 34 iterations to converge.

In Figure 4.6(a), we show the convergence of the PMPNBP with respect to the previous algorithm PAMPAS [80] which uses Gaussian mixture to represent the messages and use Gibbs sampler to perform message products (for circle). Convergence here is shown as the average error of the Maximum Likely Estimate from its ground truth with respect to the number of belief iterations over 10 trials. We plot this convergence for PMPNBP with $M = \{50, 75, 100, 200\}$ components versus PAMPAS. The convergence of PMPNBP is better than our implementation of PAMPAS. It can also be noted that the PMPNBP has decreasing average error with increasing numbers of particles. This essentially indicates that as larger M the better the inference will be. To evaluate whether PMPNBP accommodates the use of larger M in practice, we plot the CPU run time per message update iteration in Figure 4.6(b). An entire message generation in PAMPAS takes $O(KDM^2)$ operations, where D is the number of messages to compute product in the "pre-message", K is the number of iterations for the Gibbs sampler and M is the number of components representing a message. In contrast, PMPNBP takes only $O(DM)$ operations. For the plots in Figure 4.6(b) with PAMPAS we use $K = 50$ as the Gibbs sampler iterations with $D = 4$.

These results indicate that the proposed PMPNBP has similar convergence properties as the earlier approaches with greater computational efficiency.

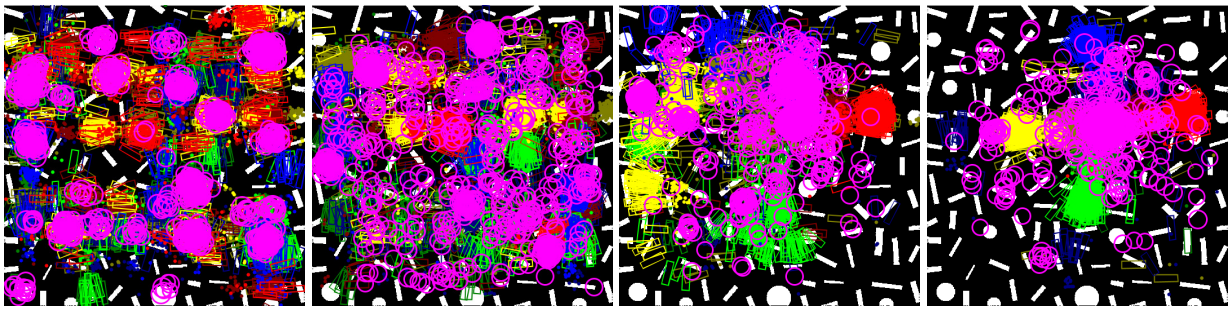
4.4.2 Real World Experiments with RGB-D Observations

We use Fetch robot, a mobile manipulation platform for our data collection and manipulation experiments. RGBD data is collected using an ASUS Xtion RGBD sensor mounted on the robot along with the intrinsic and camera to robot base transform. We use CUDA-OpenGL interoperation to render synthetic scenes on large set of poses in a single render buffer on a GPU. We render scenes as depth images, then project them back to 3D point clouds via camera intrinsic parameters.

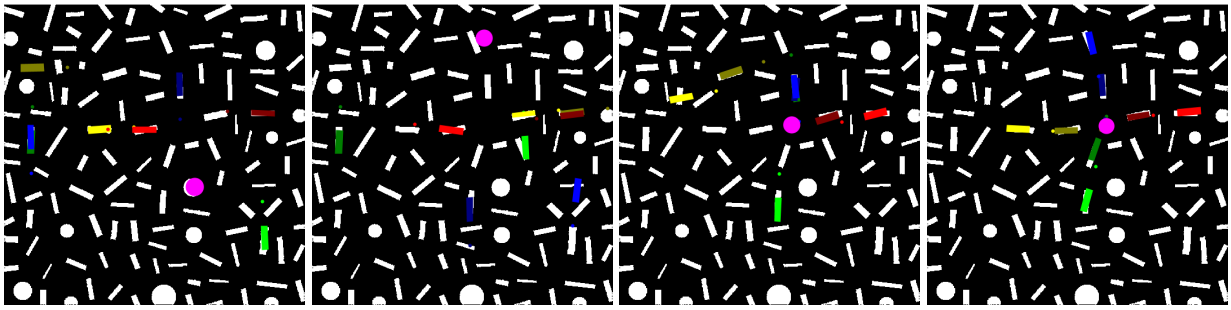
4.4.2.1 Potential Functions for Real World Experiments

Unary potential $\phi_t(X_t, Y_t)$ is used to model the likelihood by measuring how a pose X_t explains the point cloud observation P_t. The hypothesized object pose X_t is used to position the given geometric object model and generate a synthetic point cloud P_t^* that can be matched with the observation P_t. The synthetic point cloud is constructed using the object-part's geometric model available *a priori*. The likelihood is calculated as

$$\phi_t(X_t, Y_t) = e^{\lambda_r d(P_t, P_t^*)} \tag{4.11}$$

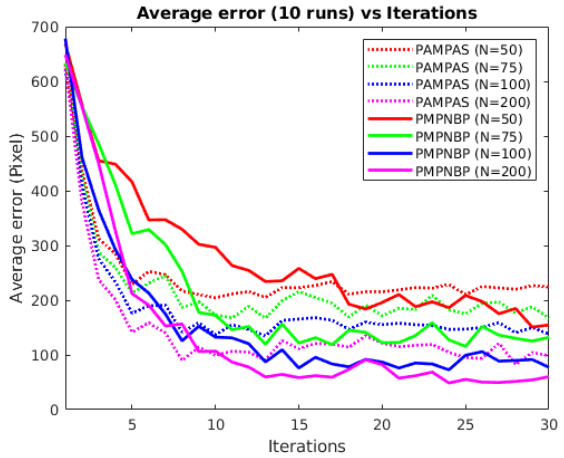


Belief at iteration 0 | Belief at iteration 1 | Belief at iteration 10 | Belief at iteration 34

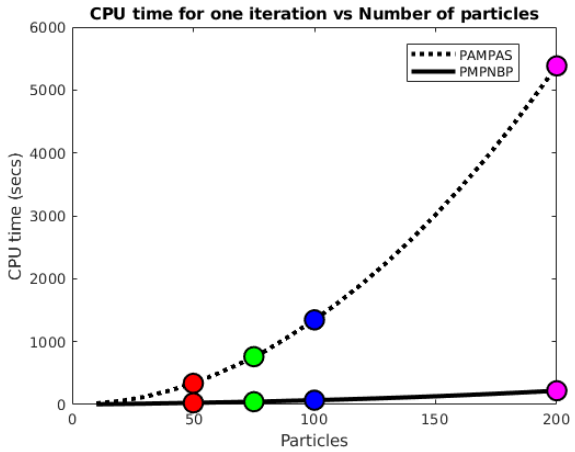


MLE at iteration 0 | MLE at iteration 1 | MLE at iteration 10 | MLE at iteration 34

Figure 4.5: Convergence in 2D articulation pose under clutter and occlusion: PMPNBP results with circle node "occluded". Each message contains 200 particles initialized randomly at locations where their $\phi_s > 0.4$. The top row shows the belief samples bel_s for each of the nodes and the bottom row shows their Maximum Likely Estimate (MLE). MLE at iteration 34 has all the links and circle converged to their ground truth states (Best viewed in color).



(a) Average error vs Iterations



(b) CPU time vs Particles

Figure 4.6: Convergence plot and execution time plot: (a) shows the average position error of Maximum Likely Estimate (MLE) achieved by PMPNBP ($M = \{50, 75, 100, 200\}$) in comparison to PAMPAS (our implementation) for the experiment in Figure 4.4. (b) shows CPU time per iteration required for PMPNBP and PAMPAS, as the number of particles grow. This shows the PMPNBP achieves comparable convergence with efficient computation.

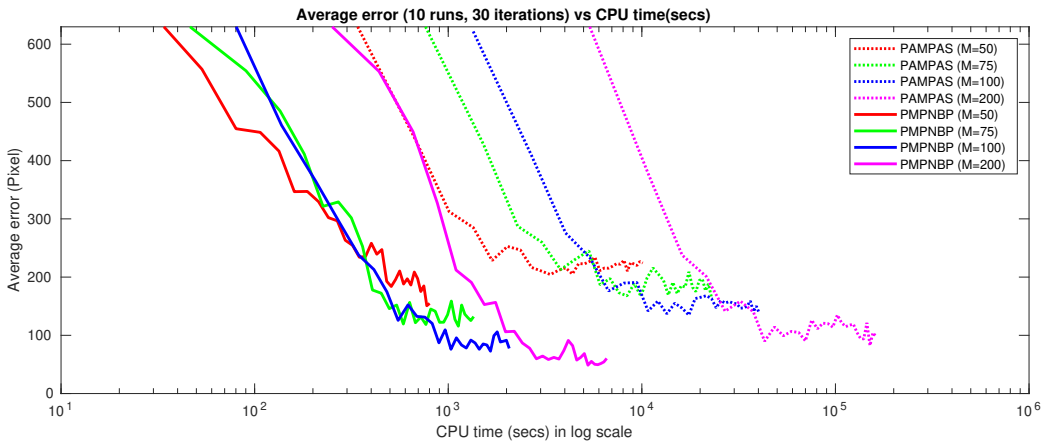


Figure 4.7: Error vs execution time plot: Shows the average position error of Maximum Likely Estimate (MLE) achieved by PMPNBP ($M = \{50, 75, 100, 200\}$) in comparison to PAMPAS (our implementation) for the experiment in Figure 4.4, versus the CPU time per iteration in the log scale.

where λ_r is the scaling factor, $d(P_t, P_t^*)$ is the sum of 3D Euclidean distance between the observed point $p \in P_t$ and rendered point $p^* \in P_t^*$ at each pixel location in the region of interest.

Pairwise potential $\psi_{t,s}(X_t|X_s)$ gives information about how compatible two object poses are given their joint articulation constraints captured by the edge between them. These constraints are captured using dual quaternions. Most often, the joint articulation constraints have minimum and maximum range in either prismatic or revolute types. We capture this information from URDF to get $R_{t|s} = [dq^a_{t|s}, dq^b_{t|s}]$ giving the limits of articulations. For a given X_s and $R_{t|s}$, we find the distance between X_t and the limits as $A = d(X_t, dq^a_{t|s})$ and $B = d(X_t, dq^b_{t|s})$, as well as the distance between the limits $C = d(dq^a_{t|s}, dq^b_{t|s})$. Using a joint limit kernel parameterized by $(\sigma_{pos}, \sigma_{ori})$, we evaluate the pairwise potential as:

$$\psi_{t,s}(X_t|X_s) = e^{-\frac{(A_{pos}+B_{pos}-C_{pos})^2}{2(\sigma_{pos})^2} - \frac{(A_{ori}+B_{ori}-C_{ori})^2}{2(\sigma_{ori})^2}} \tag{4.12}$$

The pairwise sampling uses the same limits $R_{t|s}$ to sample for X_t given a X_s. We uniformly sample a dual quaternion $\bar{X}_t$ that is between $[dq^a_{t|s}, dq^b_{t|s}]$ and transform it back to the X_s's current frame of reference by $X_t = X_s * \bar{X}_t$.

4.4.3 Articulated Objects Models

We used a cabinet with three drawers as our articulated object in the experiment. CAD model of the object is obtained from the Internet and annotation of their articulations are performed on Blender to generate URDF models. Obtaining geometrical models and articulation models can either be crowd-sourced [128] or learned using human or robot interactions [74].

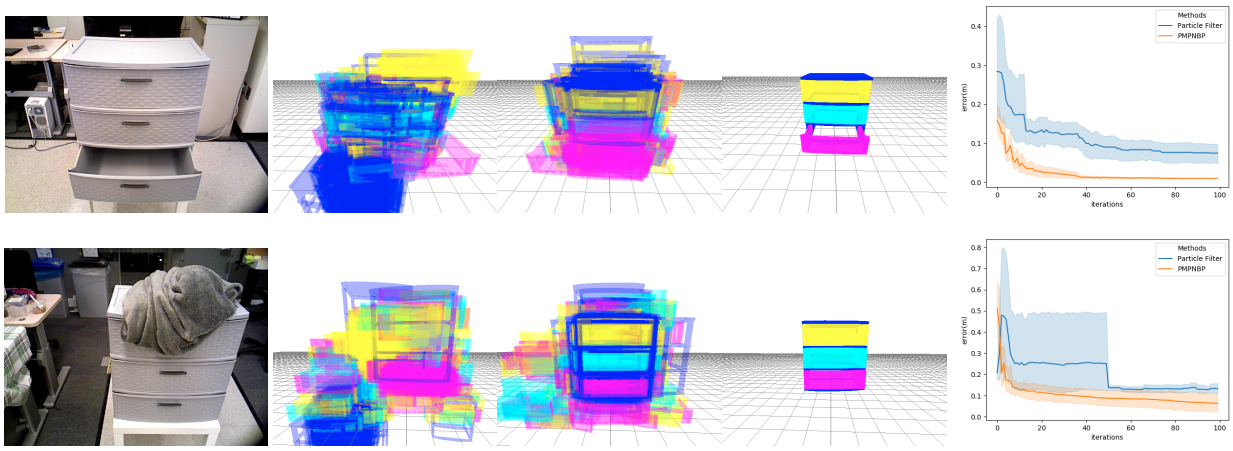


Figure 4.8: Convergence in cabinet pose and comparison with particle filter baseline. Convergence of pose estimation on two different scenes: the first column shows the RGB image of each scene, second to fourth columns show the convergence results of PMPNBP. The second column shows randomly initialized belief particles, the third column shows the belief particles after 100 iterations, and the fourth column shows the maximum likely estimates of each part. The fifth column shows the estimation error (0.95 confidence interval) using PMPNBP with respect to the baseline particle filter method across 10 runs (400 particles and 100 iterations each). It can be seen that the baseline suffers from local minimas while PMPNBP is able to recover from them effectively.

4.4.4 Baseline

We implemented Monte Carlo localization (particle filter) method that has object specific state representation. For example, the Cabinet with 3 drawers have state representation of $(x, y, z, \phi, \psi, \chi, t_a, t_b, t_c)$ where the first 6 elements describe the 6D pose of the object in the world and t_a, t_b, t_c represent the prismatic articulation. The measurement model in the implementation uses the unary potential described in the Section 4.4.2.1. Instead of rendering a point cloud of each object-part, the entire object in the hypothesized pose is rendered for measuring the likelihood. As the observations are static, the action model in the standard particle filter is replaced with a Gaussian diffusion over the object poses.

4.4.4.1 Convergence Results

In the Figure. 4.8, we show the convergence of the proposed method visually for two scenes containing different point cloud observations. We collected point cloud observations of the objects in arbitrary poses and performed inference using both the proposed PMPNBP and the baseline Monte Carlo localization. Entire point cloud observed by the sensor is used as the observation for all the object-parts. The first column shows the scene (RGB not used in the inference). Second column shows the uniformly initialized poses of the object-parts on the entire point cloud. Third column shows the propagated belief particles for each object-part after 100 iterations. Fourth col-

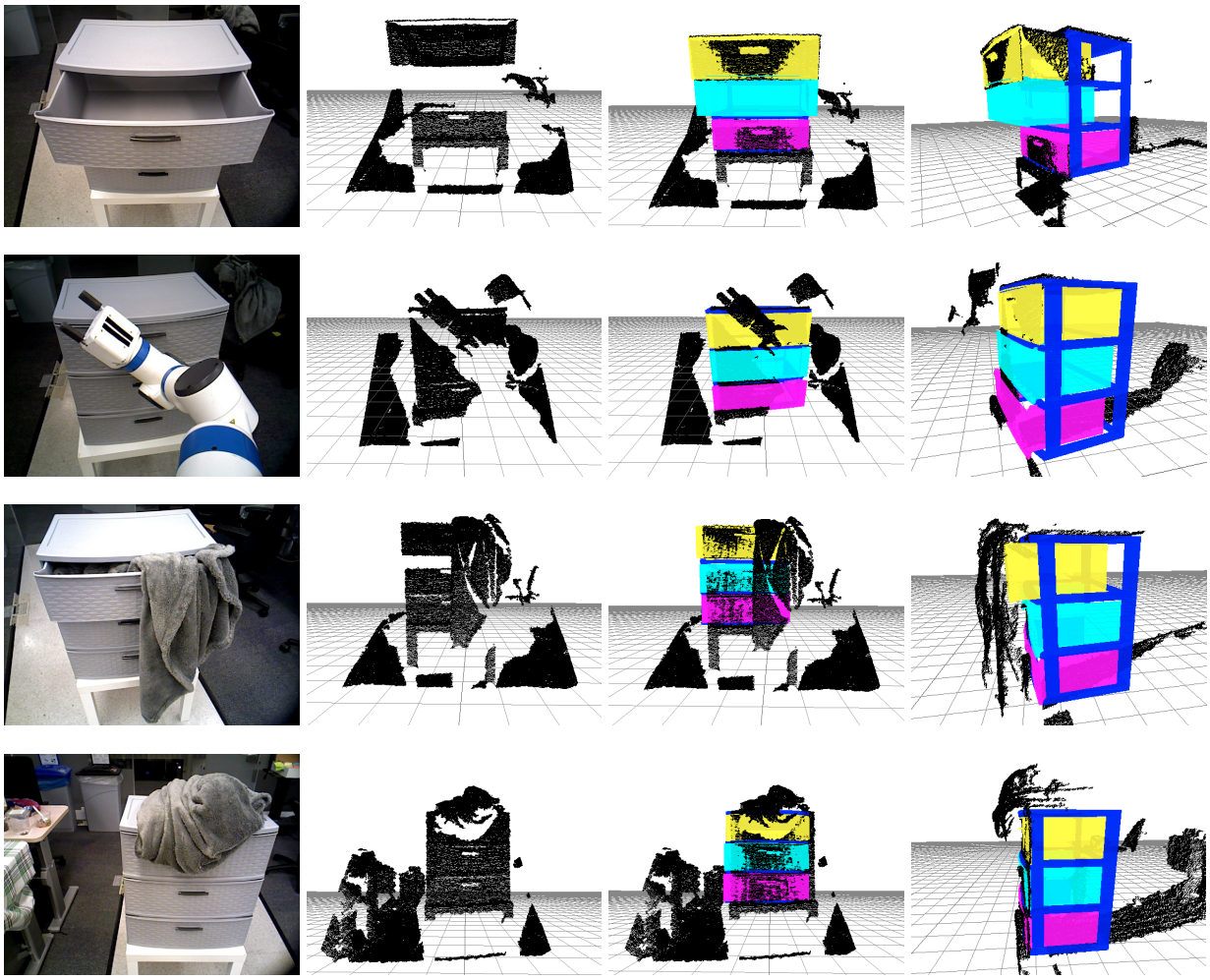
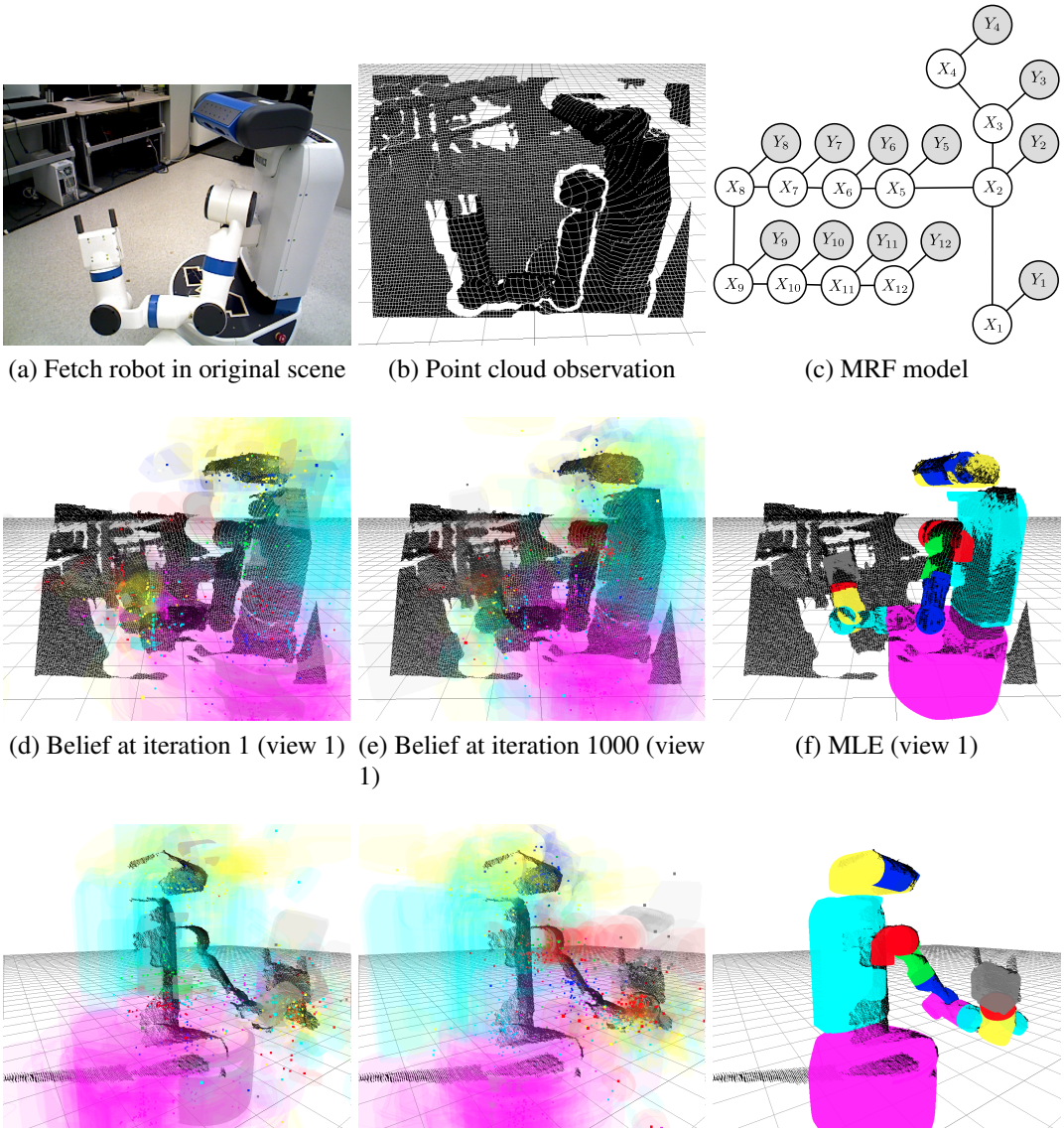


Figure 4.9: Convergence in cabinet pose under different occlusions. Partial and incomplete observations due to self and environmental occlusions are handled by PMPNBP to estimate a plausible and accurate pose

umn shows the Maximum Likely Estimate (MLE) of each object-part using the belief particles from the third column.

For the results shown in Figure. 4.8, we ran our inference for 100 iterations with 400 particles representing the messages. 10 different runs are used to generate the convergence plot that shows the mean and variance in error across the runs. We adopt the average distance metric (ADD) proposed in [129, 6] for the evaluation. The point cloud model of the object-part is transformed to its ground truth dual quaternion (dq) and to the estimated pose's dual quaternion ($\bar{dq}$). Error is calculated as the pointwise distance of these transformation pairs normalized by the number of



(a) Fetch robot in original scene (b) Point cloud observation (c) MRF model

(d) Belief at iteration 1 (view 1) (e) Belief at iteration 1000 (view 1) (f) MLE (view 1)

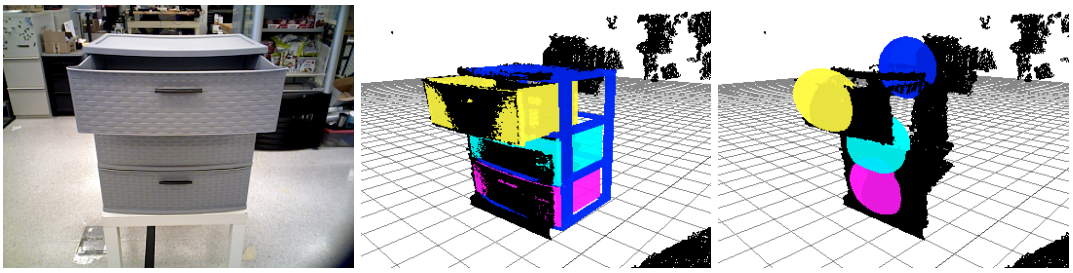
(g) Belief at iteration 1 (view 2) (h) Belief at iteration 1000 (view 2) (i) MLE (view 2)

Figure 4.10: Factored pose estimation using PMPNBP extends to articulated objects such as Fetch robot: (a) which has 12 nodes and 11 edges in the probabilistic graphical model (c). For a scene (a), which has partial 3D point cloud observation (b), the PMPNBP message passing algorithm, propagates the belief samples from iteration 1 (d and g) to iteration 1000 (e and h), that leads to MLE (f and i).

points in the model point cloud. The average distance metric (ADD) is given as

$$ADD = \frac{1}{m} \sum_{p \in \mathcal{M}} \| \bar{dq} * p * \bar{dq_c} - dq * p * dq_c \| \tag{4.13}$$

where $(\bar{dq_c})$ and (dq_c) are the conjugates of the dual quaternions [130, 131], m is the number of 3D



(a) Scene observed (b) Estimate from PMPNBP (c) Confidence ellipsoids on belief samples

(d) Grasping drawer 3 (e) Opening drawer 3

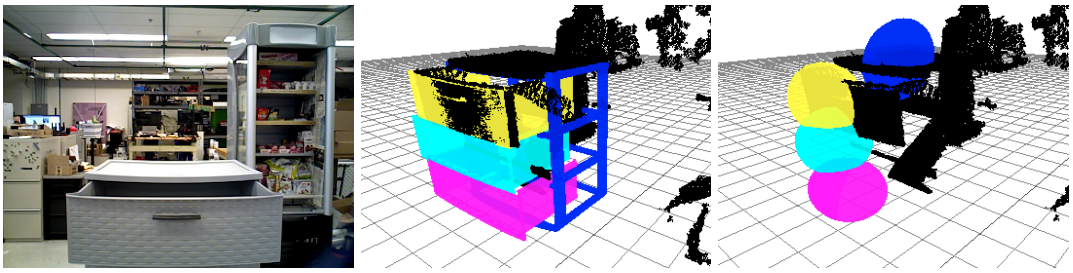
Figure 4.11: Cabinet manipulation scenario 1: The task for the robot is to open the drawer 3 (bottom) while having the drawer 1 open. The robot estimates the state of the object with certainty as shown in (b) with drawer 1 open and drawer 3 closed. In addition to the estimate, covariance can be calculated (shown as ellipsoids in (c) with 75% confidence interval). This could be used to decide that the estimation is certain with a threshold on the standard deviation on each of the dimensions of the pose. In this case the standard deviation of (x, y, z) falls below the threshold 0.25cm, and allows the robot to perform the opening action. (d-e) shows the robot performing opening action on drawer 3 using the estimate.

points in the model set $\mathcal{M}$.

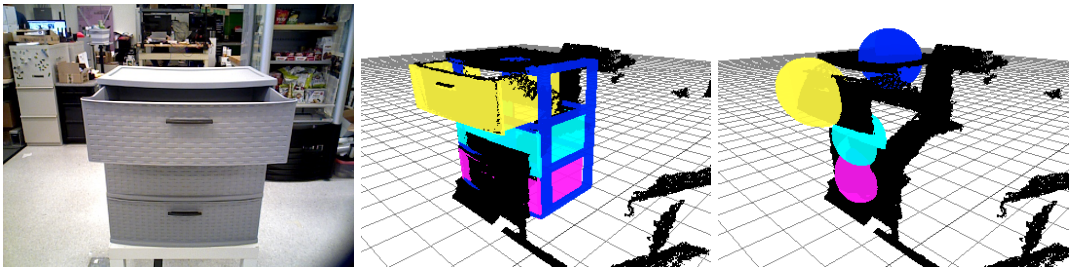
4.4.4.2 Partial and Incomplete Observations

Articulated models suffer from self-occlusions and often environmental occlusions. By exploiting the articulation constraints of an object in the pose estimation, our inference method is able to estimate a physically plausible estimate that can explain the partial or incomplete observations. In Figure. 4.9 we show three compelling cases that indicates the strength of our inference method. In the first case, the drawer 1 heavily occludes the bottom drawers resulting in limited observations on drawer 2 and 3. PMPNBP is able to estimate a plausible pose given the constraints. In the second case, the cabinet is occluded by the robot's arm, while in the third case, a blanket from the drawer 1 occludes half of the object. PMPNBP is able to recover from these occlusions and produce a plausible estimate along with belief of possible poses.

The factored approach proposed in this paper scales to objects with higher number of links and joints with combinations of articulations. This is evaluated by estimating the pose of a Fetch robot that has 12 nodes and 11 edges in its graphical model. The graphical model is constructed using the URDF model of the robot. This is shown in Figure. 4.10(c) where the robot is observed using



(a) Scene observed (b) Estimate from PMPNBP (c) Confidence ellipsoids on belief samples



(d) Scene observed after moving the camera (e) Estimate from PMPNBP (f) Confidence ellipsoids on belief samples

(g) Grasping drawer 3 (h) Opening drawer 3

Figure 4.12: Cabinet manipulation scenario 2: The task remains the same as Figure. 4.11. However the robot is not directly observing the drawer 3 (a). The inference using PMPNBP gives an estimate (b). However, the standard deviation in the position (as seen in (c) with ellipsoids of larger radii), is higher than the threshold 0.25cm. To reduce the uncertainty in the estimation, the robot takes an intermediate action (changing the viewpoint) to gain more information about the scene. This results in new scene (d). Running inference using PMPNBP on this scene gives an estimate (e) with covariance ellipsoids as shown in (f). This satisfies the threshold on the standard deviations, enabling the robot to perfom the (g-h) grasping and opening of drawer 3.

the a depth camera. Figure. 4.10(a & b) show the original scene and its point cloud observation with partial sensor data on the base, torso and the head of the robot. PMPNBP is able to estimate the pose of the robot by iteratively passing messages for 1000 iterations. Figures. 4.10(d-f) and Figures. 4.10(g-i) show the belief samples of the robot links at iteration 1 and 1000 followed by the most likely estimation (MLE) from two different view points for better visualization.

4.4.4.3 Benefits of Maintaining Belief towards Planning Actions

We show how the belief propagation approach aids in planning with a simple task illustration. Assume that the robot is performing a larger task of storing elements into the drawer 3. In a subtask, the goal is to open the drawer 3. With this setting (see Figure. 4.11) the robot is perceiving the current scene by estimating the pose of the cabinet, along with covariance on the belief for each part. We set a maximum threshold of 0.25cm on the standard deviation of (x, y, z) dimensions to decide if the estimation is certain or not. In this case, the standard deviation from the belief falls within this threshold and the robot is certain that the drawer 1 is open and drawer 3 is closed. Hence, the robot performs opening drawer 3 action. For the same task but with a different observation (see Figure. 4.12), the robot estimates the pose of the cabinet, along with its covariance. However, in this case, the robot is not certain about the estimation as the standard deviation is bigger than the threshold. This enables the robot to take an intermediate action (of lowering its torso) that provides a new observation of the cabinet. With this new observation, the robot perceives that the drawer 3 is closed with more certainty and performs an open action. This is an illustration of how the belief can be used in planning actions. More rigorous experiments with the choice of thresholds for different objects and tasks will be detailed in the future work.

4.5 Limitations

The key problem toward solving a belief propagation problem with continuous variables is a message product that takes $\mathcal{O}(M^D)$. M is the number of Gaussian mixture components used to represent the continuous value, and D is the number of incoming mixtures used to construct an outgoing mixture in the context of message passing. The methods discussed in Introduction proposed approximations to compute this product to make the nonparametric belief propagation tractable in their respective applications domains. Here, we propose another such approximation (PMPNBP) that is much more efficient and does not grow asymptotically as the other approximations proposed earlier. PMPNBP assumes that the belief of a node generates its incoming message reweighted by the constraints of its neighbors. On the other hand, state-of-the-art methods [80, 79] generated a new message from the source node to the target node by using all the other incoming messages to the source node. When the belief cannot capture samples at the true locations, PMPNBP will fail to generate an incoming message with samples at the true locations. Because PMPNBP can work with large number of samples, it always assumes that samples are available around the true locations that will be exploited in the inference. To avoid this scenario, a percentage of samples from uniform distribution can be used in addition to the samples from the belief. These samples can be considered as exploration samples. We consider this limitation to be reasonable because,

computationally, PMPNBP affords to use large number of samples as compared with other methods. In our experiments, for the 2D articulated pattern estimation, we used 50% of the samples to explore, whereas in the 3D cabinet and robot pose estimations, we used only 10% of the samples.

4.6 Summary

In this chapter, we addressed the limitations of Chapter 3, by factoring a scene state into objects and their parts for generating belief over the scene states efficiently. We proposed a new message passing scheme that uses a "pull" approach to update messages in Nonparametric Belief Propagation. Specifically, the proposed message passing scheme avoids Gibbs sampling based message products of the earlier methods and provides faster product approximations (Figure. 4.6). We show the efficiency of the proposed algorithm both in terms of its convergence properties and the computing time with respect to an earlier method PAMPAS on their 2D illustration. Furthermore, we apply PMPNBP to a real world articulated object pose estimation problem and show results successful estimation on scenes with partial observations. We compare our factored representation and message passing approach, with the standard representation and particle filtering approach. Our factored approach converges faster and consistently over several runs when compared to the standard approach.

We further illustrate the benefit of generating and maintaining belief for robot manipulation task, where an estimate and its pose covariance can inform a task planner on how confident a perceptual run was in estimating the state. PMPNBP described in this chapter, inherently can perform belief propagation over object poses on a stream of observations. This leads to object pose tracking across frames, that will be used in the goal-directed manipulation experiments of Chapter 5.

CHAPTER 5

Belief Propagation for Tracking Pose of Articulated Objects in Clutter

5.1 Introduction

In this chapter, we go back to the motivating scenario described in Chapter 1, where our goal was to enable the robot to perform goal-directed manipulation task to achieve the desired goal configuration. Specifically, we focus on the pose estimation and tracking problem induced by continuously changing scene observations. Consider the world state to be a collection of objects and their parts. By propagating the probabilistic notion of the world state, we propose to predict and correct the belief over the world state at every timestep. This approach is applicable to clutter scenarios in indoor settings, where partial observations most likely do not yield accurate estimation, and belief across timesteps should be utilized.

The message update algorithm introduced in Chapter 4 assumes that belief samples are in close proximity to the solution. In other words, the message update will only weight the hypotheses represented by belief samples from the previous iteration. To accommodate this assumption, a percentage of the belief samples are drawn from a uniform distribution over possible solution space. This augmentation helps us achieve better convergence properties. Figure 5.1, shows the accuracy in estimating the 2D pattern described in Chapter 4 with 50 particles and 100 iterations, with respect to the percentage of samples from the uniform distribution. It can be seen that at 40% percentage, the average error and the standard deviation over 100 runs are observed to be optimal. In Chapter 4, it is demonstrated that pose estimation of large objects such as cabinet and robot can be performed with the help of this augmentation to the belief samples. For applications such as tracking of objects over a sequence of observations requires the inference engine with a fixed budget in the number of particles and iterations. Additionally, if the objects to be tracked are smaller in the observation space, the inference demands discriminatively informed samples. In this chapter, we explore various methods with the objectives: 1) limiting the number of particles and iterations and 2) handling smaller objects such as handtools.

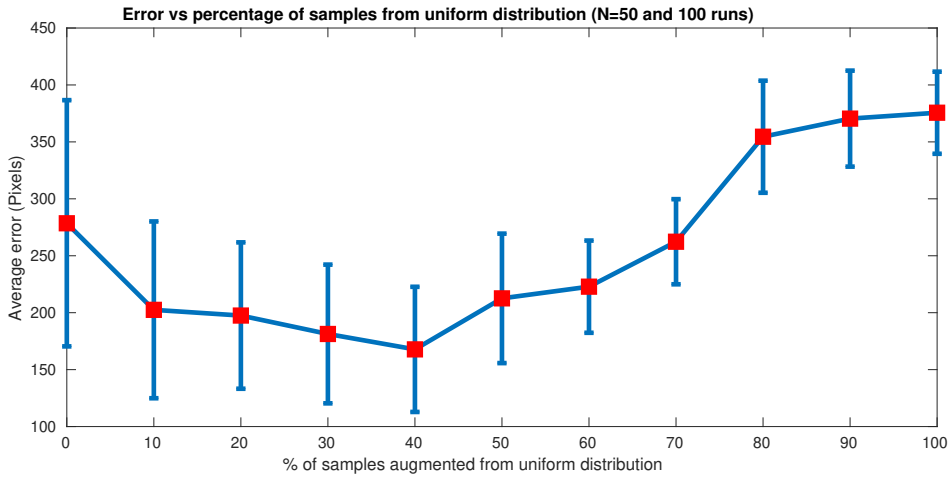


Figure 5.1: Accuracy of PMPNBP with respect to the percentage of uniformly sampled belief particles at every iteration

First, we explore max-product variants of Belief Propagation (BP) to limit the number of particles and iterations in the inference. In Section 5.2, we provide the background and existing methods on max-product algorithms. In the subsequent section, we discuss how the augmentation techniques of the max-product algorithms can be adopted to overcome the *uninformative augmentation* discussed above. Particle selection step that follows the augmentation step maintains the number of samples to yield the same number of samples throughout the inference. We describe three simple and possible ways to perform particle selection that are inspired by particle filter [110], a sum-product (PMPNBP [16, 17]), and a max-product (Diverse Particle Max-Product (D-PMP) [91]) algorithms. We compare these three selection steps that follow the augmentation step and compare their convergence and computational properties on the 2D pattern estimation described in Chapter 4.

Secondly, we describe our proposed framework to track articulated handtools under manipulation. In addition to the message passing module with augmentation and selection steps, a part-wise segmentation network is used to provide a heat-map that provides a pixel-wise probability of the appearance of an object's part. The framework takes in a sequence of RGBD images along with the 3D geometry and articulation model of the target object and produces marginal belief over the poses of the object's parts along with a MAP estimate. Qualitative results are shown on scenarios with handtools articulated by a human to describe how the proposed framework is capable of propagating belief samples under heavy occlusions during the demonstration.

5.2 Related Work and Background

Tracking objects in 3D scenes for robotic manipulation applications has garnered much interest in the robotics community in recent years. The availability of high resolution 3D depth camera data in robotics has sparked work in object pose estimation and tracking in 3D scenes [132]. A broad theme of these approaches has been to use probabilistic methods for tracking. Wuthrich et al. [65] propose a probabilistic technique for tracking of objects being manipulated by a human or robot with known geometries using a particle filter which models occlusions in addition to the observation and process models. Issac et al. [70] modify the Gaussian filter to track object models robustly and efficiently.

Various work has been done on tracking articulated objects. In [67], Schmidt et al. introduce a general framework for tracking articulated objects with known structure using an extended Kalman filter where the observation model uses the signed distance function. It was extended to include physics based constraints on the objects [68]. Makris et al. [69] propose a hierarchical model fusion framework for visual tracking in which a defined object model hierarchy guides the inference of a main model by fusing the inferences made on simpler auxiliary models.

Articulated 3D model tracking work has extensively focused on tracking hands [133] and manipulator end-effectors [134]. The methods mentioned require an estimate of the initial pose [65, 67, 68], have only been shown to work on a specific problem domain [133], or require additional sensor information beyond depth data [134]. Our proposed tracking framework differentiates itself from these in that we estimate and track object pose over time with the same framework, with no initialization required.

In Chapter 4, we discuss the existing sum-product BP algorithms and propose our efficient pull message passing algorithm for estimating pose of articulated objects. These algorithms employ particle-based approximations to the continuous BP messages. A complimentary family of algorithms to the sum-product are the max-product algorithms that focus on maximum a posteriori (MAP) inference problems. While the sum-product algorithms compute marginal distributions using important sampling and resampling techniques, max-product algorithms take a optimization perspective to find the posterior modes (see Chapter 2 for more background). Restating the problem described in Chapter 4 in Equation 4.1 with temporal component T here, we are interested in finding the X^T that maximizes the joint probability given as

$$p(X^T, Z^T) = \frac{1}{Z} \prod_{(s,t)\in E} \psi_{s,t}(X_s^T, X_t^T) \prod_{s\in V} \phi_s(X_s^T, Z_s^T). \tag{5.1}$$

Instead of Equation 5.6 in Chapter 4, the approximated message from a node $t \to s$ for the max-

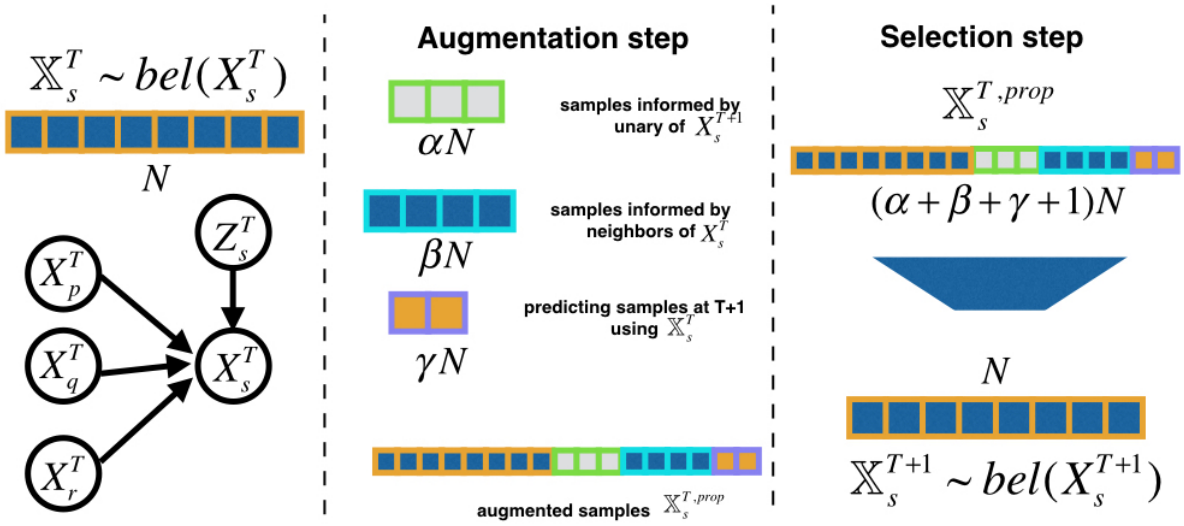


Figure 5.2: Message passing via augmentation and selection steps

product variants of the belief propagation algorithms is given by

$$\hat{m}_{t\to s}^n(X_s^T) = \max_{X_t^T \in \mathbb{X}_t^T} \psi_{ts}(X_t^T, X_s^T)\phi_t(X_t^T, Z_t^T) \prod_{u\in\rho(t)\setminus s} \hat{m}_{u\to t}^{n-1}(X_t^T), \tag{5.2}$$

where $\mathbb{X}_t^T$ is the particle set of the node t denoting its belief $bel(X_t^T)$.

Pacheco et al [91] propose the Diverse Particle Max-product algorithm (D-PMP) that is devised to preserve modes of hypotheses in problems where there is more than one solution. For example, D-PMP is applied to estimating human body pose in an RGB observation with more than one person in it. D-PMP maintains significantly better modes compared to its max-product counterparts, such as Metropolis Particle Max-Product (M-PMP) [92], Greedy Particle Max-Product (G-PMP) [93] and PatchMatch & Top-N Particle Max-Product (T-PMP) [94].

5.3 Methodology

Belief propagation via iterative message passing is a common approach to infer hidden variables while maximizing the joint probability of a graphical model. The distribution of a rigid part's pose X_s^T is represented in a nonparametric form as a set of belief particles denoted by $bel_s(X_s^T)$ where $X_s^T \in \mathbb{X}_s^T$. We adopt the max-product iterative message passing approach to perform this inference, where messages are passed between hidden variables until their beliefs converge. A message - denoted by $m_{t\to s}^n(X_s^T)$ - can be considered as a belief of the receiving node s as informed by the sender t at iteration n for timestep T. An approximation of the message - denoted as $\hat{m}_{t\to s}^n(X_s^T)$ -

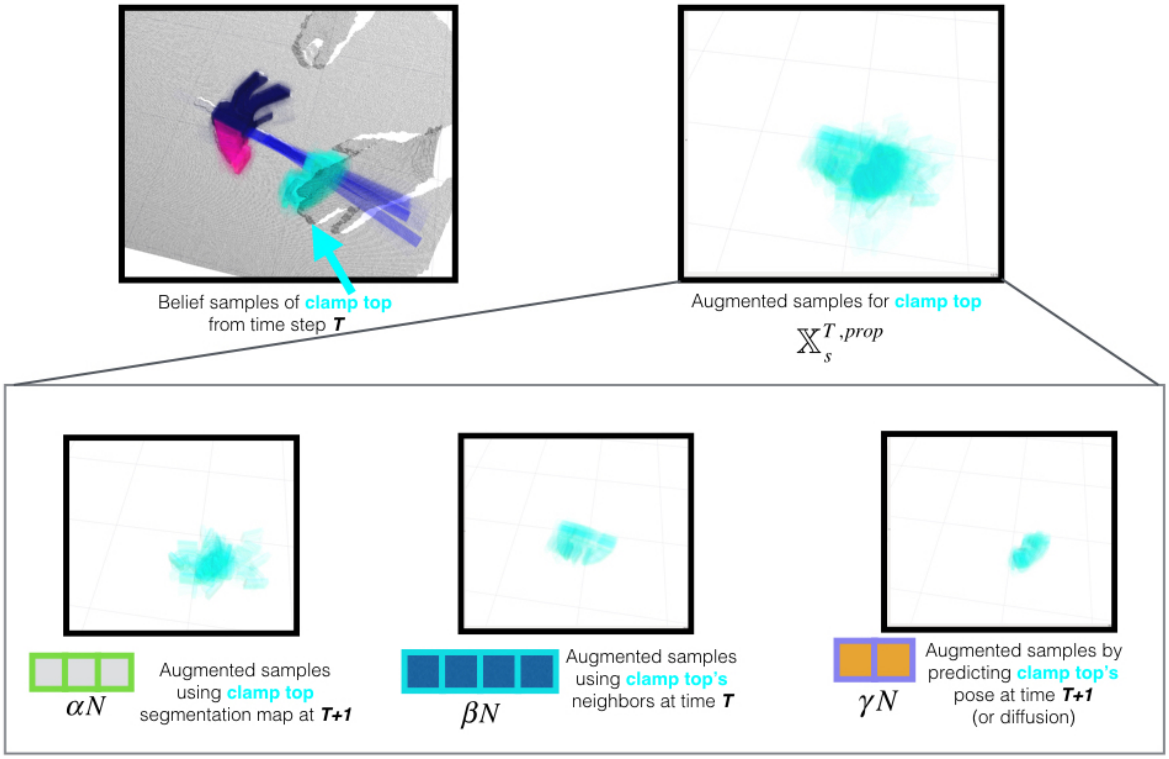


Figure 5.3: Augmentation Illustration for clamp's top part

is computed typically using the sum-product of the incoming messages as (described in Chapter 4).

$$\hat{m}^n_{t\to s}(X^T_s) = \sum_{X^T_t \in \mathbb{X}^T_t} \psi_{s,t}(X^T_s, X^T_t)\phi_t(X^T_t, Z^T_t) \prod_{u\in\rho(t)\setminus s} \hat{m}^{n-1}_{u\to t}(X^T_t), \tag{5.3}$$

where $\rho(t)$ and $\mathbb{X}^T_t$ denote the set of neighboring hidden nodes and the particle set of node t, respectively.

The marginal belief of a hidden node is a product of all the incoming messages weighted by the node's unary potential:

$$bel^n_s(X^T_s) \propto \phi_s(X^T_s) \prod_{t\in\rho(s)} \hat{m}^n_{t\to s}(X^T_s). \tag{5.4}$$

.

Our particle optimization algorithm aims to approximate the joint probability of the MRF, as in Equation 5.1, by maintaining the marginal belief, as in Equation 5.4 for each object part. Section 5.3.1 describes this message passing algorithm. Section 5.4.1 describes how the functions $\phi(X^T_s, Z^T_s)$ and $\psi(X^T_s, X^T_t)$ are modelled for the tracking experiment.

5.3.1 Belief Propagation via Message Passing

Estimating an articulated object's pose or state in terms of the 6D pose of its parts increases the size of the solution space. Additionally cluttered settings with similar parts and partial observations makes the inference prone to convergence to local minima. To mitigate this problem while computing messages, we can optionally add an augmentation step to accommodate different proposals as opposed to the traditional update and resample steps of traditional iterative particle refinement methods. We refer to [91] for this augmentation. More precisely, our method has the following steps: *augmentation*, and *selection*. Then, the augmented particles are *reweighted* and evaluated to produce N samples for the next iteration. The overall system is summarized in Figure 5.2. The following subsections describe these steps in more detail.

5.3.1.1 Augmentation Step

At each node s, the particle set representing the distribution $\mathbb{X}_s^T$ can be augmented by drawing particles from various proposal distributions. Given N particles in the distribution, Gaussian noise over the 6 DoF pose is first added to the current particles, then the distribution is augmented to $\mathbb{X}_s^{T,prop} = \mathbb{X}_s^T \cup \mathbb{X}_s^{T,aug}$ which contains $(\alpha + \beta + \gamma + 1)N$ particles, where α, β and γ denotes the fraction of particles from various proposals q_s^{pair}, q_s^{unary}, and q_s^{rand} respectively. This is illustrated in the Figure. 5.3.

Pairwise: Pairwise proposal distribution $q_s^{pair}(X_s^T) \propto \psi_{s,t}(X_s^T, \tilde{X}_t^T)$, is conditioned on a neighboring sample $\tilde{X}_t^T$, that was sampled using the weights from the unary potential of t. i.e. $\tilde{X}_t \sim \phi_t$. The weight associated with this sample is $\phi_t(\tilde{X}_t^T, Z_t^T)$.

Unary: Unary proposal distribution $q_s^{unary}(X_s^T) \propto \phi_s(X_s^T, Z_s^T)$, informs the importance sampling based on the unary potential ϕ_s.

Random: Random proposal distribution $q_s^{rand}(X_s^T) \propto \mathcal{N}(X_s^T, \Sigma)$, is adding noise to the samples. This is to avoid the belief falling into a local minima due to the high dimensionality of the orientation space, and to account for mirror symmetry in some objects.

5.3.1.2 Selection Step

Reweighting and resampling: We follow the **Belief Update** algorithm from Chapter 4 to perform the selection step. Each of the auxilary set from the proposals are normalized to weight between $[0, 1]$ to augment them under the same scale of weights and resample to produce N samples for the next iteration.

Max-product: We follow the selection step via Approximated Integer Programming as proposed by Pacheco et al.[91]. This involves creating a Message Foundation Matrix with the weights from the pairwise potential and followed by sequentially picking N samples.

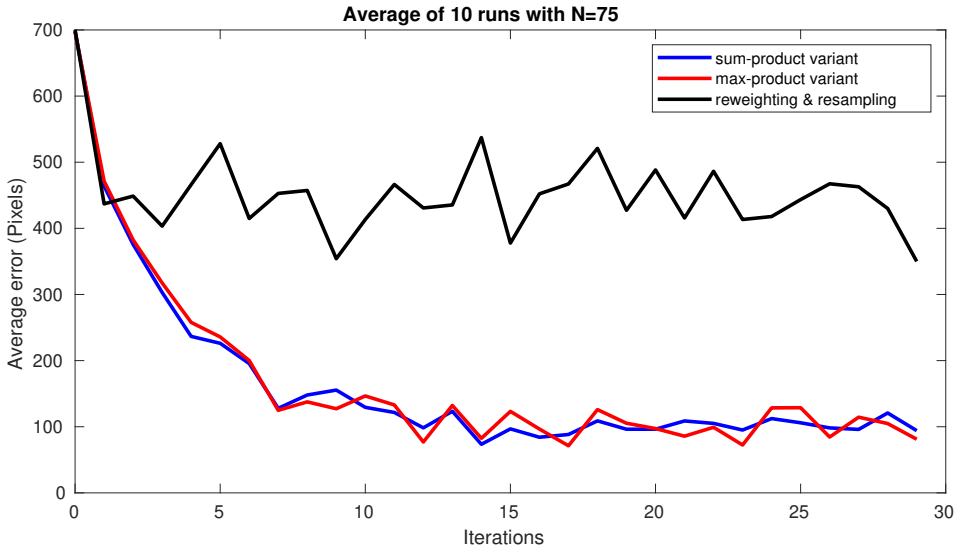


Figure 5.4: Convergence characteristics of different selection methods

Sum-product: Each particle $X_s^T \in \mathbb{X}_s^{T,prop}$ is reweighted as follows:

$$w_s = \phi_s(X_s^T, Z_s^T) \prod_{t \in \rho(s)} \hat{m}_{t \to s}^n(X_s^T), \tag{5.5}$$

where $\hat{m}_{t \to s}^n(X_s^T)$ is the sum-product message as given by the Equation. 5.3. Depending on the graph structure this only takes into account the immediate neighbours of the node, and uses the neighbour particle's unary potential $\phi_t(X_t^T, Z_t^T)$. For numerical stability, the log-likelihoods are used in practice. The weights are normalized and then the particles are resampled using importance sampling. Figure 5.4 shows the comparison between the selection methods over the 2D pattern estimation experiment described in Chapter 4. We use the sum-product variant (PMPNBP with augmentation) for the tracking experiments.

5.4 Object Tracking Experiment

Using the proposed augmentation and sum-product based selection step, we perform object tracking experiments where a sequence of RGBD observations are used to estimate the pose of an articulated object in the scene manipulated by an agent in the world. These experiments are performed on a *bar clamp*, with four parts to it: *top* and *bottom clamp jaws* to hold any objects that articulate prismatically along the axis provided by a *bar*, and a *handle* attached to the bottom to release and fix of the clamp.

We take a two-stage approach based on the success from the recent works on rigid body pose

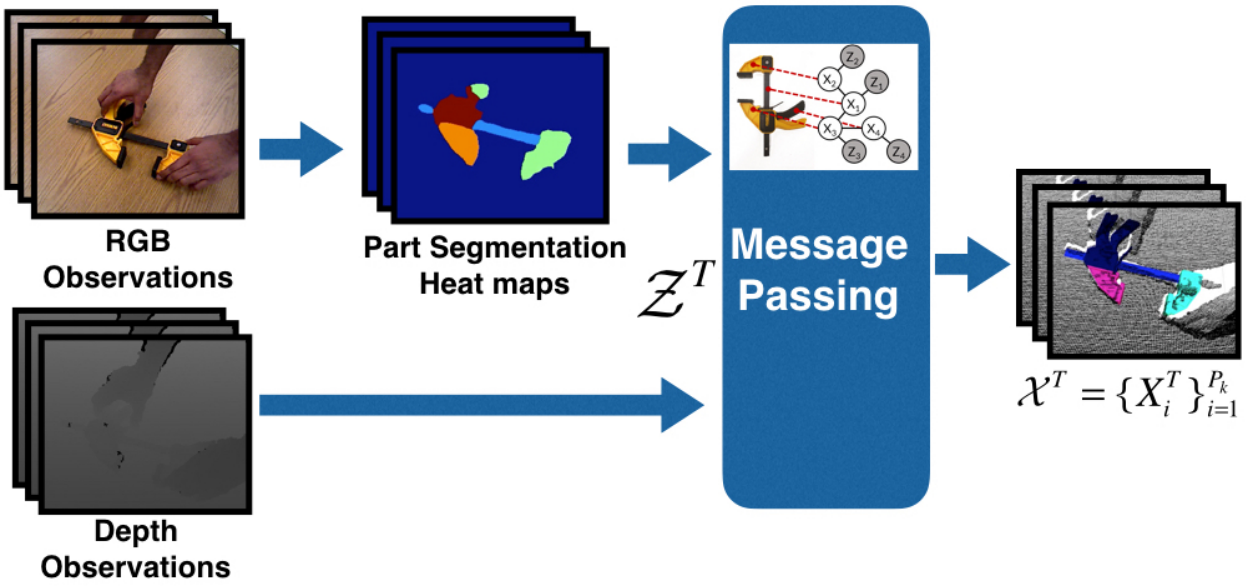


Figure 5.5: Inference pipeline for tracking the pose of articulated objects: Robot observes a scene as an RGB-D image. The RGB image is given as an input to a trained Dilated ResNets network, that generates pixel-wise heatmap for the parts of an object class of interest. The message passing is initialized with part poses using these heatmaps and the depth image, to iteratively converge to the 6D pose of each of the rigid parts.

estimation [53, 54, 55, 56, 57, 1] where the first-stage is a discriminative module that uses the sensory information to provide a prior to the inference. In our experiments, we use a segmentation module trained on handtools to provide pixel-wise probabilities on the appearance of an object part. This first stage provides a probabilistic heatmap over the pixel space, which is then used to initialize the 6D pose hypotheses of the object parts with corresponding depth observation. The trained DilatedResNet[135] used as the first stage is from work by Pavlasek et al. [136], which also provides a dataset consisting of 8 handtools with different articulations and their 3D geometries. The network is trained on $6k$ images from video sequences containing the handtools on a tabletop setting. We refer to [136] for more details on the training a part based segmentation network with articulated objects.

The potentials $\phi_t(X_t^T, Z_t^T)$ and $\psi_{t,s}(X_t^T, X_s^T)$ depend on the application. We describe the choice of these potentials in the below sections.

5.4.1 Potential Functions for Tracking Experiments

We describe the unary and pairwise potentials used in the tracking experiments. For clarity, we avoid the temporal notation T in the following subsections.

5.4.1.1 Unary Potential:

Unary potential $\phi_t(X_t, Z_t)$ is used to model the likelihood by measuring how a pose X_t explains the point cloud observation P_t. The hypothesized object pose X_t is used to position the given geometric object model and generate a synthetic point cloud P_t^* that can be matched with the observation P_t. The synthetic point cloud is constructed using the object-part's geometric model available *a priori*. In addition to the 3D point cloud information, we make use of the probabilistic heatmap coming from the segmentation module to weigh our unary potential. For a segmentation mask Z_t^{seg}, each pixel index $I = (u, v)$ has a probability for the appearance of an node t denoted as $p(Z_t, I)$.

The likelihood is calculated as

$$\phi_t(X_t, Z_t) = \frac{\text{Inlier}(P_t, P_t^*, Z_t^{seg})}{|P_t|} \times \frac{\text{Inlier}(P_t, P_t^*, Z_t^{seg})}{|P_t^*|} \tag{5.6}$$

where λ_r is the scaling factor, $\text{Inlier}(P_t, P_t^*, Z_t^{seg})$ is the Inlier function that between the observed point $p \in P_t$ and rendered point $p^* \in P_t^*$ at each pixel location in the region of interest determined by the segmentation mask. The inlier function is defined as summation over the observation space.

$$\text{Inlier}(P_t, P_t^*, Z_t^{seg}) = \sum_{i \in I} \begin{cases} c_{depth}(1 - \frac{D(p,p^*)}{\sigma}) + c_{seg} p(Z_t, i) & \text{if } D(p, p^*) < \sigma \\ p(Z_t, i)(\frac{\sigma}{D(p,p^*)}) & \text{otherwise} \end{cases}$$

In our experiemnts we use $\sigma = 0.004$ meters, $c_{depth} = 0.6$, and $c_{seg} = 0.4$.

5.4.1.2 Pairwise Potential

The pairwise likelihood between neighbouring particles $\psi_{t,s}(X_t, X_s)$ measures how compatible X_s is with respect to X_t. If X_s falls within the joint limits of s with respect to t at pose X_t, then $\psi_{t,s}(X_t, X_s) = 1$. Otherwise, the likelihood is the exponential of the negative error between X_s and the nearest joint limit. This potential is detailed in Chapter 4.

5.4.2 Tracking Results

The tracking experiments are divided by the types of interaction and occlusion in the observations: 1) occlusion without interaction, 2) occlusion during an interaction, 3) background clutter and occlusion during an interaction, and 4) occlusion during interaction with demonstrating a task. We qualitatively demonstrate the performance of estimating articulated objects under occlusion. We further discuss the limitations of the proposed work in practical settings and discuss potential solutions.

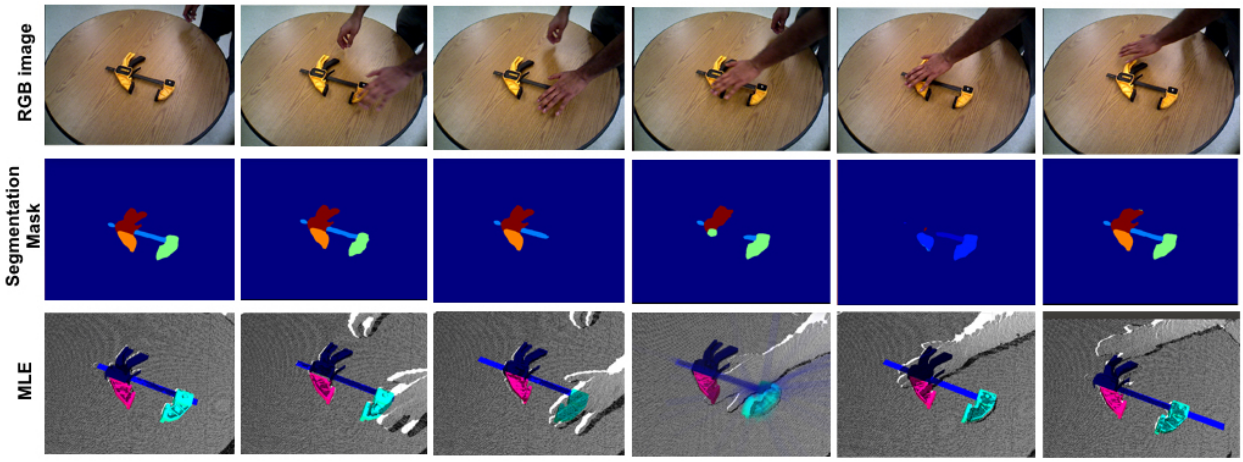


Figure 5.6: Tracking experiment with occlusion under no interaction: The first row shows the RGB observations at different frames. The second row shows the Part segmentation masks from the DilatedResNet. The third row shows the Maximum Likelihood Estimate after Message passing for 10 iterations on each of the frames.

Occlusion without interaction: In this scenario, the agent occludes the direct view of the articulated object from the sensor. This interaction produces parts of the object occluded from the sensor producing partial observations of the object in the scene. Figure 5.6 shows this scenario and the results obtained using our tracking framework. The leftmost image of Figure 5.6 shows the frame with no occlusion, resulting in a good prior from the segmentation module. The following image from left to right shows that the human agent is using his hand to occlude the parts of the articulated object. It can be seen that the object's estimation is not affected by the occlusion induced by the agent. Our inference pipeline estimates the pose of the articulated object after 10 iterations, and the Maximum likelihood estimate is shown in the bottom row. In this experiment, we use 200 particles/samples per object part.

Occlusion with interaction: In this scenario, the agent holds the articulated object and performs the articulation action. This interaction produces parts of the object occluded from the sensor producing partial observations of the object in the scene. Figure 5.7 shows this scenario and the results obtained using our tracking framework. The left-most frame in Figure 5.7 shows the frame with no occlusion, resulting in a good prior from the segmentation module. The following images show the articulation action performed by the human agent. It can be noticed that the tracking framework can track the pose of the articulated object in this sequence of frames. Our inference pipeline estimates the pose of the articulated object after 10 iterations, and the Maximum likelihood estimate is shown in the bottom row. In this experiment, we use 200 particles/samples per object part.

Background clutter with interaction: In this scenario, the articulated object is placed on a pile of tools that are from the dataset [136], and the interaction is to disturbs the pile. This interaction produces significant changes to the segments from the segmentation module. Figure 5.8 shows

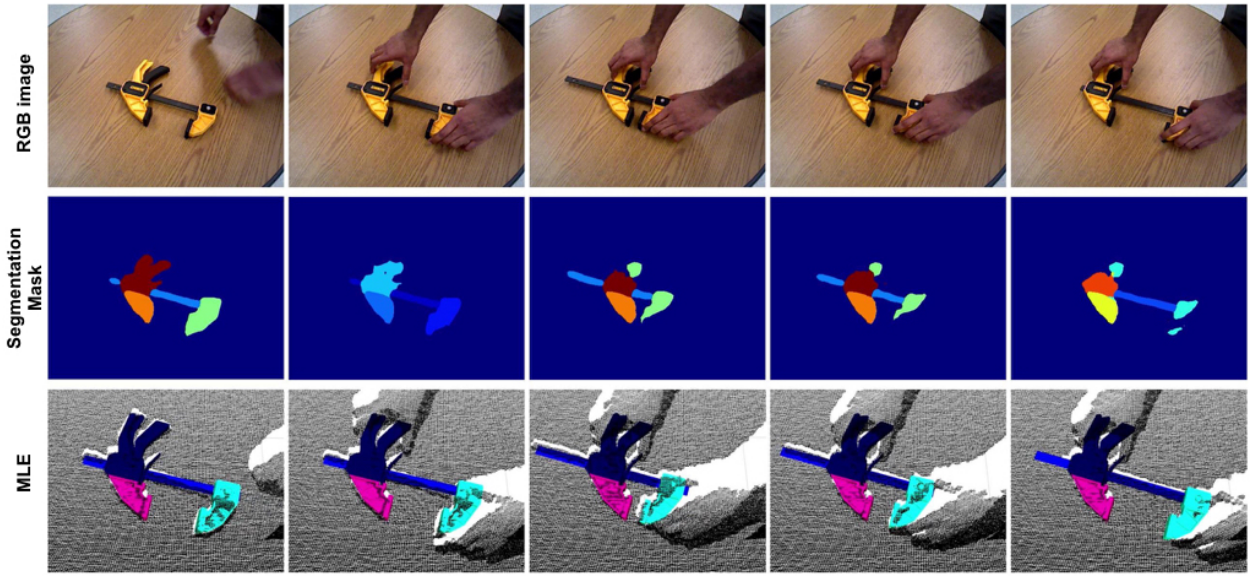


Figure 5.7: Tracking experiment with occlusion during interaction: The first row shows the RGB observations at different frames. The second row shows the Part segmentation masks from the DilatedResNet. The third row shows the Maximum Likelihood Estimate after Message passing for 10 iterations on each of the frames.

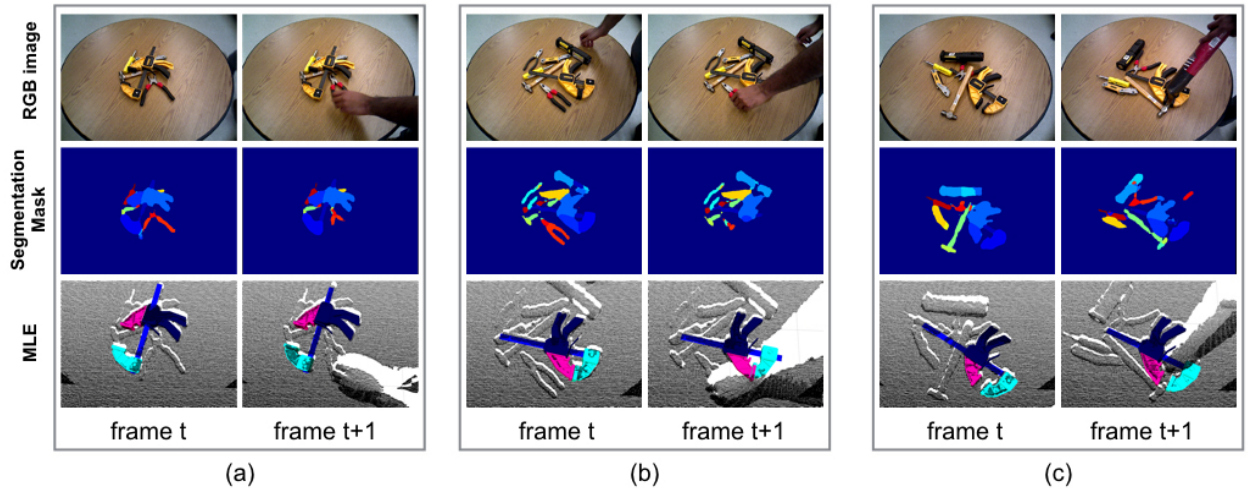


Figure 5.8: Tracking experiment with background clutter: Three experiments are shown here. The first column of each experiment is the frame preceding the second column. It can be seen that the estimates continue to persist in the right locations under the occlusion.

three scenarios (2 columns belonging to the same scene) and with before and after frames under clutter action. The first column of each experiment is the frame preceding the second column. It can be seen that the estimates continue to persist in the right locations under the occlusion. Our inference pipeline estimates the pose of the articulated object after 10 iterations, and the Maximum likelihood estimate is shown in the bottom row. In this experiment, we use 200 particles/samples per object part.

Occlusion during interaction with task demonstration: In this scenario, the human agent uses

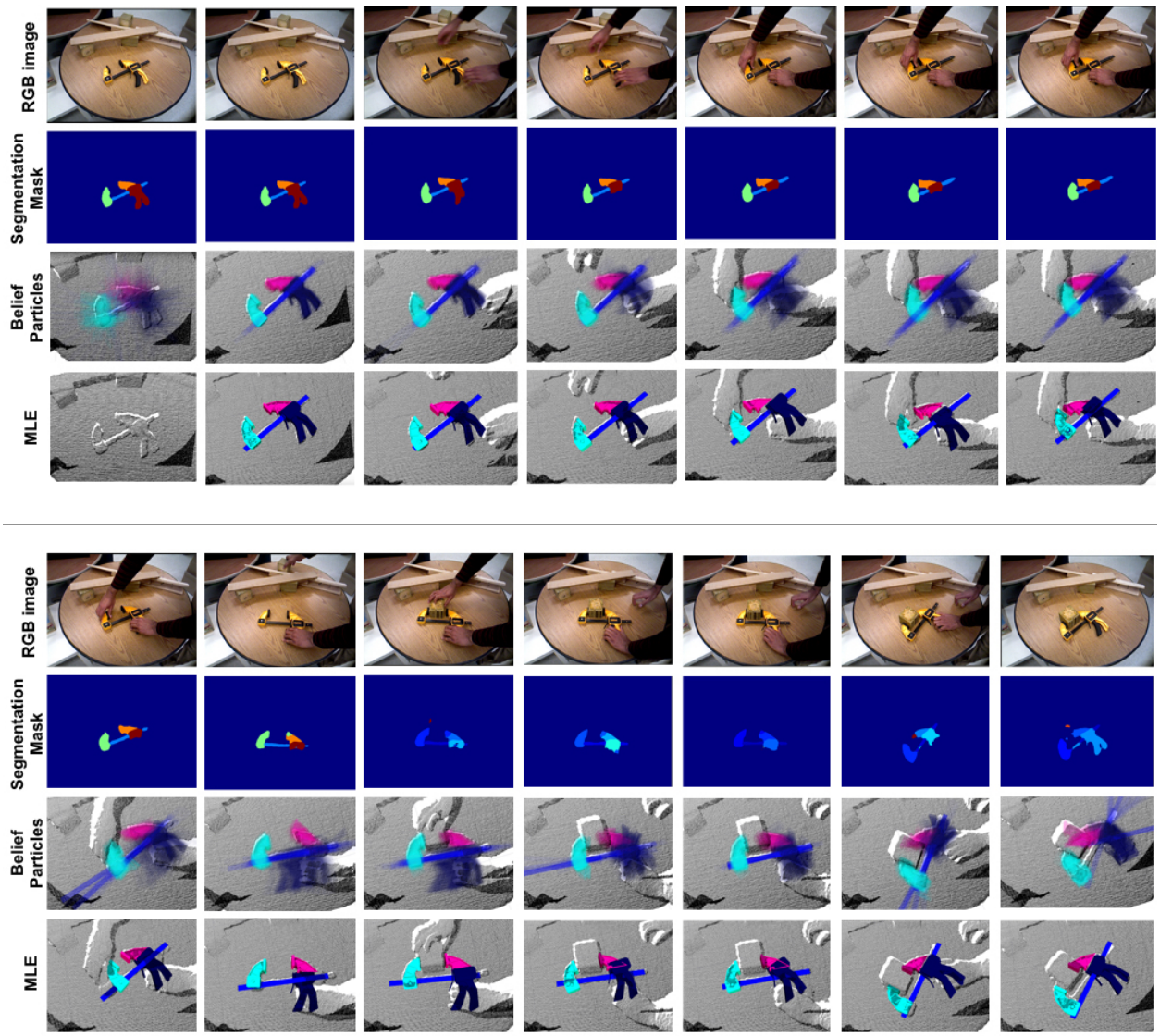


Figure 5.9: Tracking experiment with task demonstration: 14 frames from a sequence is shown here. In this sequence, the human demonstrates the clamp's articulation, followed by clamping a wooden block to it. It can be seen that during the interaction, the segmentation masks loses the information of the handle part of the object. The belief samples show the probable location of this handle in the scene, and the Maximum likelihood estimate shows the best estimate during the inference of that frame. The first frame in this figure shows the initialized belief particles with no MLE at this moment.

the clamp to perform clamping action with wooden blocks on the table. Figure 5.9 shows this scenario with the sequence of frames and their segmentations. This figure shows how the belief particles maintain a distribution over a possible pose of the part under occlusion. Our inference pipeline estimates the pose of the articulated object after 10 iterations, and the Maximum likelihood estimate is shown in the bottom row. In this experiment, we use 200 particles/samples per object part.

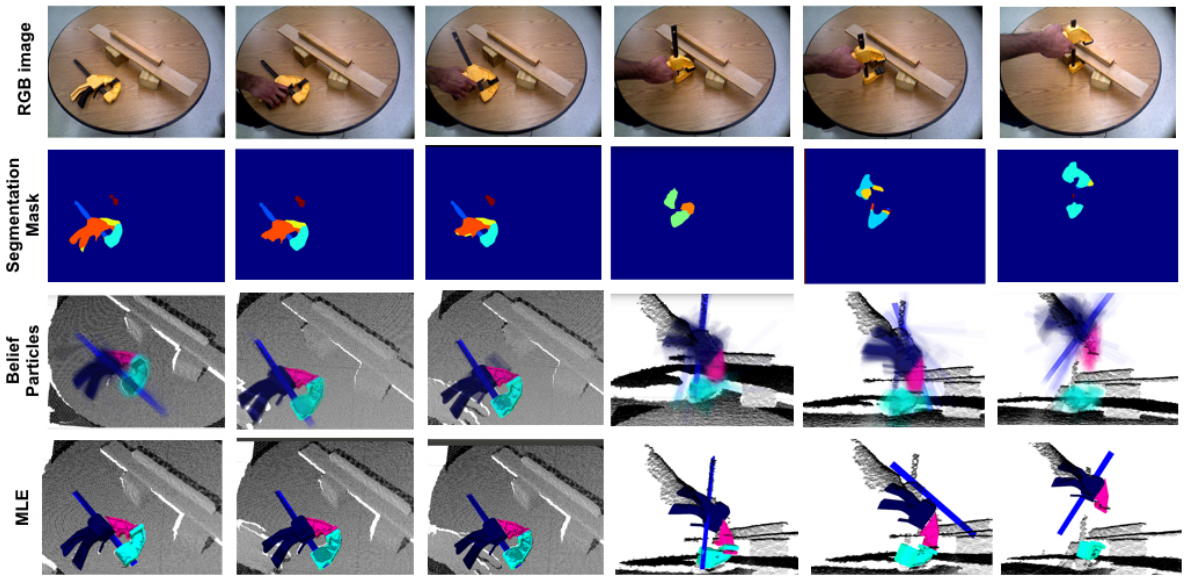


Figure 5.10: Tracking experiment showing the limitation of the tracking pipeline: In this experiment, the sequence shows a human picking up the clamp and clamping it to the wooden platform on the table. It can be seen that the clamp is estimated in the first three frames. However, from frame 4 (from the left) onwards, the segmentation mask does not include any segments on the bar, the bottom part of the clamp as well as the handle. It also misclassifies the pixels of the bottom part of the clamp as the top part. It can be seen that the belief samples tend to be close to the right locations. However, they are not converged well enough to produce a reasonable estimate.

5.5 Limitations and Failure Scenarios

The proposed message passing through augmentation and selection steps cater to the needs of the tracking framework with a limited number of particles and fewer iterations. However, it heavily relies on the segmentation module to provide good proposals in the augmentation step. Figure 5.10 shows a failed experiment, where the segmentation for more than two parts out of four parts are occluded, failing in the estimation after three frames.

Estimating the correct pose of the articulated object under severe occlusions is an under-constrained problem. There are cases where the belief over possible pose a part of the object take is spread out in the 3D space. In such scenarios, the Maximum likelihood estimate is difficult to maintain the overall structure of the object.

5.6 Summary

In this chapter, we tackled the problem of pose estimation and tracking of scene hypotheses, where a scene is composed of articulated objects under partial time-variant observations. Specifically, this chapter explores the sum-product and max-product variants of belief propagation algorithms while catering to the needs of the tracking problem. Especially to limit the number of particles

and iterations while being able to localize smaller objects with respect to the observation space. The proposed framework utilizes a segmentation network that provides pixel-wise prior about the appearance of an object part. The discriminative module's addition enhances the localization of the articulated object and tracking over continuous observations under occlusions.

CHAPTER 6

Conclusion and Future directions

Goal-directed manipulation tasks in unstructured human environments necessitate scene estimation that accommodates uncertainty due to sensing and action execution, and complies with task and motion planners. The uncertainty in sensing is predominantly due to partial observations under occlusion. Generative inference provides a way to generate and maintain a distribution over possible scene hypothesis and accommodate this uncertainty. In this thesis, we present ideas and methods that efficiently generate and maintain these hypotheses while estimating the scene as a collection of objects or their parts.

6.1 Contributions

This dissertation includes the following key contributions.

In Chapter 3, we presented a particle-based inference method that generatively estimates a scene ensuring physical plausibility. Specifically, we proposed two variants of the particle-based inference framework [15] that uses Monte Carlo sampling approaches. The developed algorithms explain partial observations with objects under heavy occlusions by producing plausible estimates in the real world.

In Chapter 4, intending to make generative inference tractable, we presented a factorization approach where a scene or an object can be factored into objects or their rigid-parts. This factorization is formulated as a Markov Random Field (MRF) and solved using Nonparametric Belief Propagation. We proposed fast inference using a Pull Message Passing algorithm (PMPNBP) [16, 17] and demonstrate its efficiency by estimating scenes with articulated objects. We demonstrated that our proposed method has a significant gain in computation compared to a state-of-the-art message passing algorithm.

In Chapter 5, to extend the message passing methods to continuous observations, we explored the sum-product and max-product variants of belief propagation algorithms while catering to the needs of the tracking problem. Specifically, we showed ways to limit the number of particles and

iterations while being able to localize smaller hand tool objects over time-varying observations. The proposed framework utilized a segmentation network that provides pixel-wise prior about the appearance of an object part. The discriminative module's addition enhanced the localization of the articulated object and tracking over continuous observations even under occlusions.

6.2 Future Directions

In this section, we present some future directions based on the ideas from this dissertation.

6.2.1 Learning the Potential Functions

In Chapters 4 and 5, we discussed unary and pairwise potential functions that drive the Belief Propagation algorithms. In our experiments, these potential functions and their modeling plays a crucial role to scale these algorithms to a wide range of objects and scenes while keeping the inference tractable. It is desired to learn these potential functions from large amounts of data. A recent work by Pavlasek et al. [136] explores this direction to study the effects of a learned unary potential. Sigal et al. [14] learned the potential functions describing the structure of a human to estimate and track the pose of a human over a sequence of observations. Recent work by Cao et al. [137], proposes affinity fields, that could potentially act as a pairwise potential for message passing algorithms to generalize across objects. However, the challenge here is to scale these learning-based methods to a wide range of objects in the human environment.

6.2.2 Part and Affordance Discovery during Interaction

The methods proposed in this dissertation assumes that the 3D geometry of objects and their articulations to be known apriori. This assumption hurdles the general applicability of these methods in practice. Extracting precise 3D geometry using a sequence of frames has been studied since the advent of RGBD cameras. Maghoumi et al. [138] explored intuitive ways to extract the 3D geometry from sketch-based interfaces, paving ways to think about part-based geometries and their affordances from the perspective of human users. Recent works such as DynamicFusion [139] have shown promising results in continually estimating and tracking dynamic deformable geometries such as humans. As an extension of the ideas in Chapter 5, it is necessary to look into ways to relax the assumption of the 3D geometry and estimate the 3D structure of the object while learning to use the object toward a specific task.

BIBLIOGRAPHY

[1] Zeng, Z., Zhou, Z., Sui, Z., and Jenkins, O. C., "Semantic Robot Programming for Goal-Directed Manipulation in Cluttered Scenes," *IEEE/RSJ International Conference on Robotics and Automation (ICRA)*, 2018.

[2] Fikes, R. E. and Nilsson, N. J., "STRIPS: A new approach to the application of theorem proving to problem solving," *Artificial intelligence*, Vol. 2, No. 3, 1972, pp. 189–208.

[3] Winograd, T., "Understanding natural language," *Cognitive psychology*, Vol. 3, No. 1, 1972, pp. 1–191.

[4] Sucan, I. A. and Chitta, S., "MoveIt!" .

[5] Beeson, P. and Ames, B., "TRAC-IK: An open-source library for improved solving of generic inverse kinematics," *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on*, IEEE, 2015, pp. 928–935.

[6] Xiang, Y., Schmidt, T., Narayanan, V., and Fox, D., "Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes," *arXiv preprint arXiv:1711.00199*, 2017.

[7] Ren, S., He, K., Girshick, R., and Sun, J., "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE transactions on pattern analysis and machine intelligence*, Vol. 39, No. 6, 2017, pp. 1137–1149.

[8] Johnson, A. E. and Hebert, M., "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Transactions on pattern analysis and machine intelligence*, Vol. 21, No. 5, 1999, pp. 433–449.

[9] Rusu, R. B., Blodow, N., and Beetz, M., "Fast point feature histograms (FPFH) for 3D registration," *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, IEEE, 2009, pp. 3212–3217.

[10] Aldoma, A., Tombari, F., Rusu, R. B., and Vincze, M., "OUR-CVFH–oriented, unique and repeatable clustered viewpoint feature histogram for object recognition and 6DOF pose estimation," *Joint DAGM (German Association for Pattern Recognition) and OAGM Symposium*, Springer, 2012, pp. 113–122.

[11] Rusu, R. B., Bradski, G., Thibaux, R., and Hsu, J., "Fast 3d recognition and pose using the viewpoint feature histogram," *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, IEEE, 2010, pp. 2155–2162.

[12] Narayanan, V. and Likhachev, M., "Discriminatively-guided Deliberative Perception for Pose Estimation of Multiple 3D Object Instances." *Robotics: Science and Systems*, 2016.

[13] Sui, Z., Xiang, L., Jenkins, O. C., and Desingh, K., "Goal-directed robot manipulation through axiomatic scene estimation," *The International Journal of Robotics Research*, Vol. 36, No. 1, 2017, pp. 86–104.

[14] Sigal, L., Bhatia, S., Roth, S., Black, M. J., and Isard, M., "Tracking loose-limbed people," *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, Vol. 1, IEEE, 2004, pp. I–I.

[15] Desingh, K., Jenkins, O. C., Reveret, L., and Sui, Z., "Physically plausible scene estimation for manipulation in clutter," *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*, 2016.

[16] Desingh, K., Lu, S., Opipari, A., and Jenkins, O. C., "Factored pose estimation of articulated objects using efficient nonparametric belief propagation," *2019 International Conference on Robotics and Automation (ICRA)*, IEEE, 2019, pp. 7221–7227.

[17] Desingh, K., Lu, S., Opipari, A., and Jenkins, O. C., "Efficient nonparametric belief propagation for pose estimation and manipulation of articulated objects," *Science Robotics*, Vol. 4, No. 30, 2019.

[18] Coradeschi, S. and Saffiotti, A., "An introduction to the anchoring problem," *Robotics and Autonomous Systems*, Vol. 43, No. 2, 2003, pp. 85–96.

[19] Harnad, S., "The symbol grounding problem," *Physica D: Nonlinear Phenomena*, Vol. 42, No. 1, 1990, pp. 335–346.

[20] Kuipers, B., "The spatial semantic hierarchy," *Artificial Intelligence*, Vol. 119, No. 1, 2000, pp. 191–233.

[21] Rusu, R. B., Marton, Z. C., Blodow, N., Dolha, M., and Beetz, M., "Towards 3D point cloud based object maps for household environments," *Robotics and Autonomous Systems*, Vol. 56, No. 11, 2008, pp. 927–941.

[22] Herbst, E., Ren, X., and Fox, D., "RGB-D object discovery via multi-scene analysis," *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, IEEE, 2011, pp. 4850–4856.

[23] Nakanishi, J., Morimoto, J., Endo, G., Cheng, G., Schaal, S., and Kawato, M., "Learning from demonstration and adaptation of biped locomotion," *Robotics and autonomous systems*, Vol. 47, No. 2-3, 2004, pp. 79–91.

[24] Akgun, B., Cakmak, M., Jiang, K., and Thomaz, A. L., "Keyframe-based learning from demonstration," *International Journal of Social Robotics*, Vol. 4, No. 4, 2012, pp. 343–355.

[25] Chernova, S. and Veloso, M., "Interactive policy learning through confidence-based autonomy," *Journal of Artificial Intelligence Research*, Vol. 34, 2009, pp. 1–25.

[26] Grollman, D. H. and Jenkins, O. C., "Incremental Learning of Subtasks from Unsegmented Demonstration," *International Conference on Intelligent Robots and Systems (IROS 2010)*, Taipei, Taiwan, Oct 2010, pp. 261–266.

[27] Veeraraghavan, H. and Veloso, M., "Teaching sequential tasks with repetition through demonstration," *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 3*, International Foundation for Autonomous Agents and Multiagent Systems, 2008, pp. 1357–1360.

[28] Chao, C., Cakmak, M., and Thomaz, A. L., "Towards grounding concepts for transfer in goal learning from demonstration," *Proceedings of the Joint IEEE International Conference on Development and Learning and on Epigenetic Robotics (ICDL-EpiRob)*, IEEE, 2011.

[29] Yang, Y., Li, Y., Fermüller, C., and Aloimonos, Y., "Robot Learning Manipulation Action Plans by" Watching" Unconstrained Videos from the World Wide Web." *AAAI*, 2015, pp. 3686–3693.

[30] Mohan, S., Mininger, A. H., Kirk, J. R., and Laird, J. E., "Acquiring grounded representations of words with situated interactive instruction," *Advances in Cognitive Systems*, Citeseer, 2012.

[31] Kirk, J. R. and Laird, J. E., "Learning Task Formulations through Situated Interactive Instruction," *Proceedings of the Second Annual Conference on Advances in Cognitive Systems*, 2013, pp. 219–236.

[32] Wintermute, S. and Laird, J. E., "Bimodal Spatial Reasoning with Continuous Motion," *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 3*, AAAI'08, AAAI Press, 2008, pp. 1331–1337.

[33] Narayanaswamy, S., Barbu, A., and Siskind, J. M., "A visual language model for estimating object pose and structure in a generative visual domain," *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, IEEE, 2011, pp. 4854–4860.

[34] Tenorth, M. and Beetz, M., "KnowRob: A Knowledge Processing Infrastructure for Cognition-enabled Robots," *Int. J. Rob. Res.*, Vol. 32, No. 5, April 2013, pp. 566–590.

[35] Quigley, M., Conley, K., Gerkey, B. P., Faust, J., Foote, T., Leibs, J., Wheeler, R., and Ng, A. Y., "ROS: an open-source Robot Operating System," *ICRA Workshop on Open Source Software*, 2009.

[36] Srivastava, S., Riano, L., Russell, S., and Abbeel, P., "Using Classical Planners for Tasks with Continuous Operators in Robotics," *Proceedings of the ICAPS Workshop on Planning and Robotics (PlanRob)*, 2013.

[37] Rusu, R. B., Marton, Z. C., Blodow, N., Dolha, M., and Beetz, M., "Towards 3D Point Cloud Based Object Maps for Household Environments," *Robot. Auton. Syst.*, Vol. 56, No. 11, Nov. 2008, pp. 927–941.

[38] Ciocarlie, M., Hsiao, K., Jones, E. G., Chitta, S., Rusu, R. B., and Şucan, I. A., "Towards reliable grasping and manipulation in household environments," *Experimental Robotics*, Springer Berlin Heidelberg, 2014, pp. 241–252.

[39] Rosman, B. and Ramamoorthy, S., "Learning Spatial Relationships Between Objects," *Int. J. Rob. Res.*, Vol. 30, No. 11, Sept. 2011, pp. 1328–1342.

[40] Collet, A., Berenson, D., Srinivasa, S. S., and Ferguson, D., "Object Recognition and Full Pose Registration from a Single Image for Robotic Manipulation," *IEEE International Conference on Robotics and Automation (ICRA), Kobe, Japan*, 2009.

[41] Papazov, C., Haddadin, S., Parusel, S., Krieger, K., and Burschka, D., "Rigid 3D geometry matching for grasping of known objects in cluttered scenes," *The International Journal of Robotics Research*, 2012.

[42] Cosgun, A., Hermans, T., Emeli, V., and Stilman, M., "Push planning for object placement on cluttered table surfaces," *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 4627–4632.

[43] Pas, A. T. and Platt, R., "Localizing Handle-like Grasp Affordances in 3D Point Clouds," *International Symposium on Experimental Robotics (ISER)*, 2014.

[44] Mahler, J., Liang, J., Niyaz, S., Laskey, M., Doan, R., Liu, X., Ojea, J. A., and Goldberg, K., "Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics," *Robotics: Science and Systems (RSS)*, 2017.

[45] Joho, D., Tipaldi, G. D., Engelhard, N., Stachniss, C., and Burgard, W., "Nonparametric Bayesian Models for Unsupervised Scene Analysis and Reconstruction," *Proceedings of Robotics: Science and Systems*, Sydney, Australia, July 2012.

[46] Dogar, M. and Srinivasa, S., "A Framework for Push-Grasping in Clutter," *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, June 2011.

[47] Dogar, M. R., Hsiao, K., Ciocarlie, M. T., and Srinivasa, S. S., "Physics-Based Grasp Planning Through Clutter," *Robotics: Science and Systems*, 2012.

[48] Zhang, L. and Trinkle, J., "The application of particle filtering to grasping acquisition with visual occlusion and tactile sensing," *ICRA*, 2012.

[49] Sui, Z., Jenkins, O. C., and Desingh, K., "Axiomatic Particle Filtering for Goal-directed Robotic Manipulation," *IROS, 2015*.

[50] Zhou, Z., Sui, Z., and Jenkins, O. C., "Plenoptic monte carlo object localization for robot grasping under layered translucency," *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 1–8.

[51] Tremblay, J., To, T., Sundaralingam, B., Xiang, Y., Fox, D., and Birchfield, S., "Deep Object Pose Estimation for Semantic Robotic Grasping of Household Objects," *Conference on Robot Learning (CoRL)*, 2018.

[52] Wang, C., Xu, D., Zhu, Y., Martín-Martín, R., Lu, C., Fei-Fei, L., and Savarese, S., "Densefusion: 6d object pose estimation by iterative dense fusion," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3343–3352.

[53] Narayanan, V. and Likhachev, M., "Deliberative object pose estimation in clutter," *IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2017, pp. 3125–3130.

[54] Mitash, C., Boularias, A., and Bekris, K., "Robust 6d object pose estimation with stochastic congruent sets," *arXiv preprint arXiv:1805.06324*, 2018.

[55] Deng, X., Mousavian, A., Xiang, Y., Xia, F., Bretl, T., and Fox, D., "PoseRBPF: A Rao-Blackwellized Particle Filter for 6D Object Pose Tracking," *arXiv preprint arXiv:1905.09304*, 2019.

[56] Chen, X., Chen, R., Sui, Z., Ye, Z., Liu, Y., Bahar, R., and Jenkins, O. C., "GRIP: Generative Robust Inference and Perception for Semantic Robot Manipulation in Adversarial Environments," *arXiv preprint arXiv:1903.08352*, 2019.

[57] Sui, Z., Zhou, Z., Zeng, Z., and Jenkins, O. C., "SUM: Sequential Scene Understanding and Manipulation," *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, IEEE, 2017.

[58] Felzenszwalb, P. F., Girshick, R. B., Mcallester, D., and Ramanan, D., "Object Detection with Discriminatively Trained Part Based Models," *IEEE transactions on pattern analysis and machine intelligence*, Vol. 32, 2009, pp. 1627–1645.

[59] Felzenszwalb, P. F. and Huttenlocher, D. P., "Pictorial structures for object recognition," *International journal of computer vision*, Vol. 61, No. 1, 2005, pp. 55–79.

[60] Xiang, Y. and Savarese, S., "Estimating the aspect layout of object categories," *IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2012, pp. 3410–3417.

[61] Mo, K., Zhu, S., Chang, A. X., Yi, L., Tripathi, S., Guibas, L. J., and Su, H., "PartNet: A Large-scale Benchmark for Fine-grained and Hierarchical Part-level 3D Object Understanding," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[62] Lu, C., Su, H., Li, Y., Lu, Y., Yi, L., Tang, C. K., and Guibas, L. J., "Beyond Holistic Object Recognition: Enriching Image Understanding with Part States," *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 6955–6963.

[63] Yi, L., Huang, H., Liu, D., Kalogerakis, E., Su, H., and Guibas, L., "Deep Part Induction from Articulated Object Pairs," *SIGGRAPH Asia*, 2018.

[64] Sudderth, E. B., Mandel, M. I., Freeman, W. T., and Willsky, A. S., "Visual hand tracking using nonparametric belief propagation," *IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04)*, 2004, pp. 189–189.

[65] Wuthrich, M., Pastor, P., Kalakrishnan, M., Bohg, J., and Schaal, S., "Probabilistic object tracking using a range camera," *IEEE International Conference on Intelligent Robots and Systems*, IEEE, November 2013, pp. 3195–3202.

[66] Cifuentes, C. G., Issac, J., Wüthrich, M., Schaal, S., and Bohg, J., "Probabilistic articulated real-time tracking for robot manipulation," *IEEE Robotics and Automation Letters*, Vol. 2, No. 2, 2016, pp. 577–584.

[67] Schmidt, T., Newcombe, R. A., and Fox, D., "DART: Dense Articulated Real-Time Tracking," *Robotics: Science and Systems X, University of California, Berkeley, USA, July 12-16, 2014*, 2014.

[68] Schmidt, T., Hertkorn, K., Newcombe, R., Marton, Z., Suppa, M., and Fox, D., "Depth-based tracking with physical constraints for robot manipulation," *2015 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2015, pp. 119–126.

[69] Makris, A., Kosmopoulos, D. I., Perantonis, S. J., and Theodoridis, S., "A hierarchical feature fusion framework for adaptive visual tracking," *Image and Vision Computing*, Vol. 29, 2011, pp. 594–606.

[70] Issac, J., Wüthrich, M., Cifuentes, C. G., Bohg, J., Trimpe, S., and Schaal, S., "Depth-based object tracking using a robust gaussian filter," *2016 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2016, pp. 608–615.

[71] Salas-Moreno, R. F., Newcombe, R. A., Strasdat, H., Kelly, P. H. J., and Davison, A. J., "SLAM++: Simultaneous Localisation and Mapping at the Level of Objects," *2013 IEEE Conference on Computer Vision and Pattern Recognition*, June 2013, pp. 1352–1359.

[72] Bohg, J., Hausman, K., Sankaran, B., Brock, O., Kragic, D., Schaal, S., and Sukhatme, G. S., "Interactive Perception: Leveraging Action in Perception and Perception in Action," *IEEE Transactions on Robotics*, Vol. 33, No. 6, Dec 2017, pp. 1273–1291.

[73] Hausman, K., Niekum, S., Osentoski, S., and Sukhatme, G. S., "Active articulation model estimation through interactive perception," *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 3305–3312.

[74] Martin, R. M. and Brock, O., "Online interactive perception of articulated objects with multi-level recursive estimation based on task-specific priors," *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, September 14-18, 2014*, 2014, pp. 2494–2501.

[75] Sturm, J., Stachniss, C., and Burgard, W., "A Probabilistic Framework for Learning Kinematic Models of Articulated Objects," *J. Artif. Intell. Res.*, Vol. 41, 2011, pp. 477–526.

[76] Sturm, J., *Approaches to Probabilistic Model Learning for Mobile Manipulation Robots*, Springer Tracts in Advanced Robotics (STAR), Springer, 2013.

[77] Li, X., Wang, H., Yi, L., Guibas, L., Abbott, A. L., and Song, S., "Category-Level Articulated Object Pose Estimation," *arXiv preprint arXiv:1912.11913*, 2019.

[78] Michel, F., Krull, A., Brachmann, E., Yang, M. Y., Gumhold, S., and Rother, C., "Pose Estimation of Kinematic Chain Instances via Object Coordinate Regression." *BMVC*, 2015, pp. 181–1.

[79] Sudderth, E. B., Ihler, A. T., Freeman, W. T., and Willsky, A. S., "Nonparametric belief propagation," *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, IEEE, 2003.

[80] Isard, M., "PAMPAS: Real-valued graphical models for computer vision," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003, pp. 613–620.

[81] Thrun, S., Burgard, W., and Fox, D., *Probabilistic robotics*, Vol. 1, MIT press Cambridge, 2000.

[82] Stachniss, C., *Exploration and Mapping with Mobile Robots*, Ph.D. thesis, University of Freiburg, Department of Computer Science, April 2006.

[83] Tokdar, S. T. and Kass, R. E., "Importance sampling: a review," *Wiley Interdisciplinary Reviews: Computational Statistics*, Vol. 2, No. 1, 2010, pp. 54–60.

[84] Liu, J. S., *Monte Carlo strategies in scientific computing*, Springer Science & Business Media, 2008.

[85] Gordon, N. J., Salmond, D. J., and Smith, A. F. M., "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings F - Radar and Signal Processing*, Vol. 140, No. 2, 1993, pp. 107–113.

[86] Kennedy, J. and Eberhart, R., "Particle swarm optimization," *Proceedings of ICNN'95-International Conference on Neural Networks*, Vol. 4, IEEE, 1995, pp. 1942–1948.

[87] Pacheco, J., *Variational Approximations with Diverse Applications*, Ph.D. thesis, Brown University Dept. of Computer Science, May 2016.

[88] Ihler, A. and McAllester, D., "Particle belief propagation," *Artificial Intelligence and Statistics*, 2009, pp. 256–263.

[89] Wainwright, M. J., Jaakkola, T. S., and Willsky, A. S., "MAP estimation via agreement on trees: message-passing and linear programming," *IEEE Transactions on Information Theory*, Vol. 51, No. 11, 2005, pp. 3697–3717.

[90] Wainwright, M. J., Jordan, M. I., et al., "Graphical models, exponential families, and variational inference," *Foundations and Trends® in Machine Learning*, Vol. 1, No. 1–2, 2008, pp. 1–305.

[91] Pacheco, J., Zuffi, S., Black, M., and Sudderth, E., "Preserving modes and messages via diverse particle selection," *International Conference on Machine Learning*, 2014, pp. 1152–1160.

[92] Kothapa, R., Pacheco, J., Sudderth, E., et al., "Max-product particle belief propagation," *Master's project report, Brown University Dept. of Computer Science*, 2011.

[93] Trinh, H. and McAllester, D., "Unsupervised learning of stereo vision with monocular cues," *Proc. of the British Machine Vision Conf.(BMVC)*, Vol. 4, Citeseer, 2009.

[94] Besse, F., Rother, C., Fitzgibbon, A., and Kautz, J., "Pmbp: Patchmatch belief propagation for correspondence field estimation," *International Journal of Computer Vision*, Vol. 110, No. 1, 2014, pp. 2–13.

[95] Choi, C. and Christensen, H. I., "RGB-D object tracking: A particle filter approach on GPU," *IROS, 2013*.

[96] "PR2 Interactive Manipulation," http://wiki.ros.org/pr2_interactive_manipulation.

[97] Vondrak, M., Sigal, L., and Jenkins, O., "Physical Simulation for Probabilistic Motion Tracking," *CVPR*, 2008.

[98] Vondrak, M., Sigal, L., and Jenkins, O. C., "Dynamical simulation priors for human motion tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 35, No. 1, 2013, pp. 52–65.

[99] Khan, Z., Balch, T., and Dellaert, F., "An MCMC-based particle filter for tracking multiple interacting targets," *ECCV 2004*.

[100] Dogar, M., Hsiao, K., Ciocarlie, M., and Srinivasa, S., "Physics-based grasp planning through clutter," 2012.

[101] Brubaker, M. A., Fleet, D. J., and Hertzmann, A., "Physics-based person tracking using the anthropomorphic walker," *International Journal of Computer Vision*, 2010.

[102] Wu, J., Yildirim, I., Lim, J. J., Freeman, B., and Tenenbaum, J., "Galileo: Perceiving physical object properties by integrating a physics engine with deep learning," *Advances in Neural Information Processing Systems*, 2015, pp. 127–135.

[103] Jia, Z., Gallagher, A. C., Saxena, A., and Chen, T., "3d reasoning from blocks to stability," *IEEE transactions on pattern analysis and machine intelligence*, Vol. 37, No. 5, 2015, pp. 905–918.

[104] Liu, Z., Chen, D., Wurm, K. M., and von Wichert, G., "Table-top scene analysis using knowledge-supervised MCMC," *Robotics and Computer-Integrated Manufacturing*, Vol. 33, 2015, pp. 110–123.

[105] Joho, D., Tipaldi, G. D., Engelhard, N., Stachniss, C., and Burgard, W., "Nonparametric Bayesian models for unsupervised scene analysis and reconstruction," *Robotics*, 2013, pp. 161.

[106] Kulkarni, T. D., Kohli, P., Tenenbaum, J. B., and Mansinghka, V., "Picture: A probabilistic programming language for scene perception," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4390–4399.

[107] Zhang, L. E. and Trinkle, J. C., "The application of particle filtering to grasping acquisition with visual occlusion and tactile sensing," *ICRA, 2012*.

[108] Narayanan, V. and Likhachev, M., "PERCH: Perception via Search for Multi-Object Recognition and Localization," *ICRA*, 2016.

[109] Collet, A., Martinez, M., and Srinivasa, S. S., "The MOPED framework: Object recognition and pose estimation for manipulation," *IJRR*, 2011.

[110] Thrun, S., Burgard, W., and Fox, D., *Probabilistic robotics*, MIT press, 2005.

[111] Hastings, W. K., "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, Vol. 57, No. 1, 1970, pp. 97–109.

[112] Blender Foundation, "Blender," .

[113] "Bullet Physics Library," www.bulletphysics.org.

[114] Besl, P. J. and McKay, N. D., "Method for registration of 3-D shapes," *Sensor Fusion IV: Control Paradigms and Data Structures*, Vol. 1611, International Society for Optics and Photonics, 1992, pp. 586–607.

[115] Mitra, N. J., Gelfand, N., Pottmann, H., and Guibas, L., "Registration of point cloud data from a geometric optimization perspective," *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, ACM, 2004, pp. 22–31.

[116] Rusinkiewicz, S. and Levoy, M., "Efficient variants of the ICP algorithm," *3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference on*, IEEE, 2001, pp. 145–152.

[117] Segal, A., Haehnel, D., and Thrun, S., "Generalized-icp." *Robotics: science and systems*, Vol. 2, 2009, p. 435.

[118] Hernandez, C., Bharatheesha, M., Ko, W., Gaiser, H., Tan, J., van Deurzen, K., de Vries, M., Van Mil, B., van Egmond, J., Burger, R., et al., "Team delft's robot winner of the amazon picking challenge 2016," *Robot World Cup*, Springer, 2016, pp. 613–624.

[119] Rusu, R. B. and Cousins, S., "3D is here: Point Cloud Library (PCL)," *ICRA*, 2011.

[120] Mitash, C., Boularias, A., and Bekris, K. E., "Improving 6d pose estimation of objects in clutter via physics-aware monte carlo tree search," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2018, pp. 1–8.

[121] Katz, D., Orthey, A., and Brock, O., "Interactive Perception of Articulated Objects," *Experimental Robotics - The 12th International Symposium on Experimental Robotics, ISER 2010, December 18-21, 2010, New Delhi and Agra, India*, 2010, pp. 301–315.

[122] Murphy, K. P., Weiss, Y., and Jordan, M. I., "Loopy belief propagation for approximate inference: An empirical study," *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, 1999, pp. 467–475.

[123] Song, L., Gretton, A., Bickson, D., Low, Y., and Guestrin, C., "Kernel Belief Propagation," *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, 2011, pp. 707–715.

[124] Ren, S., He, K., Girshick, R., and Sun, J., "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *Advances in Neural Information Processing Systems 28*, edited by C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Curran Associates, Inc., 2015, pp. 91–99.

[125] Girshick, R., Donahue, J., Darrell, T., and Malik, J., "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[126] Desingh, K., Opipari, A., and Jenkins, O. C., "Pull Message Passing for Nonparametric Belief Propagation," *arXiv preprint arXiv:1807.10487*, 2018.

[127] Chua, J. and Felzenszwalb, P. F., "Scene Grammars, Factor Graphs, and Belief Propagation," *arXiv preprint arXiv:1606.01307*, 2016.

[128] Gouravajhala, S. R., Yim, J., Desingh, K., Huang, Y., Jenkins, O. C., and Lasecki, W. S., "EURECA: Enhanced Understanding of Real Environments via Crowd Assistance," 2018.

[129] Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., and Navab, N., "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes," *Asian conference on computer vision*, Springer, 2012, pp. 548–562.

[130] Gilitschenski, I., Kurz, G., Julier, S. J., and Hanebeck, U. D., "A new probability distribution for simultaneous representation of uncertain position and orientation," *Information Fusion (FUSION), 2014 17th International Conference on*, IEEE, 2014, pp. 1–7.

[131] Kenwright, B., "A beginners guide to dual-quaternions: what they are, how they work, and how to use them for 3D character hierarchies," 2012.

[132] Newcombe, R. A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A. J., Kohli, P., Shotton, J., Hodges, S., and Fitzgibbon, A., "KinectFusion: Real-time dense surface mapping and tracking," *2011 10th IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2011*, IEEE, oct 2011, pp. 127–136.

[133] Qian, C., Sun, X., Wei, Y., Tang, X., and Sun, J., "Realtime and Robust Hand Tracking from Depth," *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, 2014, pp. 1106–1113.

[134] Cifuentes, C. G., Issac, J., Wüthrich, M., Schaal, S., and Bohg, J., "Probabilistic Articulated Real-Time Tracking for Robot Manipulation," *arXiv*, Vol. abs/1610.04871, 2016.

[135] Yu, F., Koltun, V., and Funkhouser, T., "Dilated Residual Networks," *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[136] Pavlasek, J., Lewis, S., Desingh, K., and Jenkins, O. C., "Parts-Based Articulated Object Localization in Clutter using Belief Propagation," *Under Review*, 2020.

[137] Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y., "Realtime multi-person 2d pose estimation using part affinity fields," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7291–7299.

[138] Maghoumi, M., LaVioia, J. J., Desingh, K., and Jenkins, O. C., "GemSketch: Interactive Image-Guided Geometry Extraction from Point Clouds," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2018, pp. 2184–2191.

[139] Newcombe, R. A., Fox, D., and Seitz, S. M., "Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time," *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 343–352.