Counter-Hypothetical Evidential Reasoning for Mobile Manipulation Robots

by

Elizabeth Olson

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Robotics)
in the University of Michigan
2024

Doctoral Committee:

Professor Odest Chadwicke Jenkins, Chair
Professor Jessy Grizzle
Assistant Professor Katherine Skinner
Associate Professor Leia Stirling

There are many faces to the mask of uncertainty

—Richard Bellman

Elizabeth Olson

lizolson@umich.edu

ORCID iD: 0000-0001-8156-338X

To my parents

Acknowledgments

There are many people to thank for their support during my graduate studies. Thank you to my advisor, Prof. Odest Chadwicke Jenkins, for his guidance in developing my ability to contribute to robotics through both research and service. I greatly appreciate your mentorship and advocacy. Thank you to my committee members – Prof. Jessy Grizzle, Prof. Leia Stirling, and Prof. Katherine Skinner – for their feedback and insights on my dissertation.

Thank you to many of the members of the larger robotics community for their time and efforts in helping me. I have had the opportunity to work with other faculty during my time at Michigan – Prof. Matthew Johnson-Roberson, Prof. David Fouhey, and Prof. Emily Mower Provost – all of whom have given me invaluable mentorship and advice to become the researcher I am today. A sincere thank you to Denise Edmund, who has gone above and beyond with her support and encouragement. I would also like to acknowledge the funding and other sponsorship of my studies that I received from Ford, Amazon, the College of Engineering, and the Robotics Department.

Prof. Odest Chadwicke Jenkins and I would also like to thank Prof. Ella Atkins, the founder and architect of the Michigan Robotics Graduate Program. The completion of this dissertation would not have been possible without the stewardship and wisdom of Professor Atkins. The good academic culture of the Michigan Robotics Institute was a product of the values of sincerity, respect, integrity, and equity that Professor Atkins embodies everyday. Professor Atkins guidance was crucial to helping so many students at Michigan overcome great challenges and find their path to professional success and personal purpose. This dissertation is no exception. It can be all too easy to take for granted the contributions, sacrifices, and growth mindset of good stewards such as Professor Atkins, especially in our modern era of research as a credential. As such, it is especially important to respect the architects who have paved the way for the prosperity of future generations.

Finally, thank you to all of my family and friends. I am especially lucky to have befriended Alia, Anthony, Cynthia, Daphna, Eva, Jana, Justin, and Victoria. Thank you to my family for all of their constant love and support. To my partner, Grant, thank you for always believing in me and caring for me. I would not have been able to do it without you. Completing this dissertation has meant a lot to me, but finding you in this journey has meant more.

TABLE OF CONTENTS

List of Figures

FIGURE

Abstract

Although robots can perform in structured environments, they struggle to perceive and operate within cluttered, dynamic, and previously unseen settings. Nonparametric Bayesian inference has the potential to address these problems caused by uncertainty and reason over nonlinear high dimensional states. However, sampling-based Bayesian inference methods are susceptible to mode collapse of the belief distribution, where the inference process incorrectly converges to a single region of the state space. Inference with more samples can improve the ability to represent the state space fully, but it is often not feasible due to computational and resource constraints in robotics domains.

This dissertation introduces a counter-hypothetical approach to evidential reasoning for addressing the problems of mode collapse in nonparametric Bayesian inference. Evidential reasoning, in the context of nonparametric Bayesian inference, allows us to explicitly model likelihood, ambiguity, and doubt in the underlying belief of the distribution. With these more delineated measurements of belief, we no longer need to infer quantities of ambiguity and doubt from likelihood weightings alone. We demonstrate this extension can enable nonparametric Bayesian inference to sample over high-dimensional state spaces with more robustness to particle deprivation.

We begin by introducing the **Counter-Hypothetical Particle Filter**, *CH-PF*, to overcome mode collapse when tracking rigid objects from monocular video observations. Previous methods predict failures during inference based on the likelihood function. We observe that low likelihood weightings for a given hypothesis can be attributed to error in the pose or ambiguity in the observation. For this reason, we present the

counter-hypothetical likelihood function to estimate doubt independently of likelihood or ambiguity. The counter-hypothetical particle filter quantifies the evidence that supports a hypothesis and refutes the hypothesis. This independent and explicit modeling of doubt through a proposed counter-hypothetical likelihood function enables the filter to better detect failure modes and adaptively redistribute probability mass to a null hypothesis.

To better evaluate the performance of our methods on tracking high-dimensional states under heavy occlusion, we present a benchmark dataset. The **Progress LUMBER (Looking Upon a Moving BipEdal Robot) Dataset** contains 100 sequences of a bipedal humanoid robot Digit moving within highly obstructed scenes. The annotations require no external markers to label the pose of 29 links. The occlusions featured in the dataset make it unique in its representation of real-world environments that humanoid mobile manipulators may face in practical scenarios.

Extending counter-hypothetical reasoning to higher dimensional systems, we present **Weighted And Graphical Evidential Reasoning for Differentiable Nonparametric Belief Propagation**, *(WAGER-DNBP)*. This method models evidential reasoning within a differentiable nonparametric belief propagation algorithm. WAGER-DNBP not only learns the unary and pairwise potentials via labeled tracking data but also the counter-hypothetical likelihood. We then use inconsistencies between a given hypothesis's likelihood and counter-hypothetical likelihood scores and observation to estimate ambiguity. WAGER-DNBP then uses these measurements of ambiguity to determine which observations within the factor graph should carry more weight in the posterior belief distribution. We validate our method on the Progress LUMBER Dataset to show that the explicit modeling of ambiguity and doubt within WAGER-DNBP can enable it to recover from particle deprivation more efficiently.

Chapter 1

Introduction

1.1 Motivation: Robots for the Real World

Our current times are a catalyst for the transition of robots from performing predictable jobs in repeatable factory workspaces to helping with everyday tasks in common human environments. The next generation of robots has been envisioned to assist with cooking, cleaning, organizing, and assembling on our behalf. We often avoid executing these duties ourselves due to the physical demands, time constraints, or tediousness we may associate with a given task. Beyond convenience, many societal necessities, such as infrastructure maintenance or working within hazardous areas, can be dangerous for humans even to attempt.

Extending the responsibilities of robots from repeatedly grasping the same object on a conveyor belt to finding your kitchen scissors demands an evolution of their fundamental abilities. Within the domestic realm, a realistic kitchen and its manipulation requirements are shown in Figure 1.1. Most of the objects are only partially

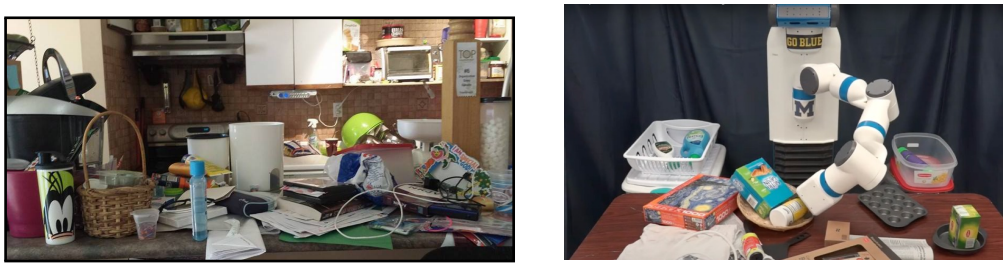


Figure 1.1: Example of the everyday environments [1] a mobile manipulator will have to understand in order to perform manipulation tasks [2].

Figure 1.2: A warehouse environment with several colocated Locus robots and humans, all of which a robot must perceive and avoid (left). Estimating the pose of another agent must be robust to partial observability, as shown in the heavy occlusion of a Figure robot (right).

observable, so the robot must form a thorough and consistent scene understanding despite only gleaming pieces of information from each vantage point or relative positioning of the objects. Such visual tracking is further complicated when the objects are moving and have high degrees of freedom. A robot might also operate within a complex and fast-paced warehouse, as depicted in Figure 1.2. However, planning its trajectory within the space carries additional safety and time constraints due to the presence of human coworkers. In all of these examples for perception and trajectory generation in robotics applications, the autonomous system must be able to reliably reason under uncertainty quickly for complex problems that provide minimal and noisy information.

Let us highlight specifically the implications of partial observability for pose tracking. In Figure 1.3, the end of a plastic banana is shown peaking out from behind a blue pitcher. The geometry of the object contains symmetry, and distinguishing between symmetries is not aided by the lack of any texture or features on the plastic. At the robot's first and partial glance at the banana, it may mistakenly flip the pose of the banana in its initial estimate. Later in the sequence, as shown in at the bottom, more of the banana is viewable. From this vantage point, it is easier for the robot to correctly estimate the pose of the banana. This example

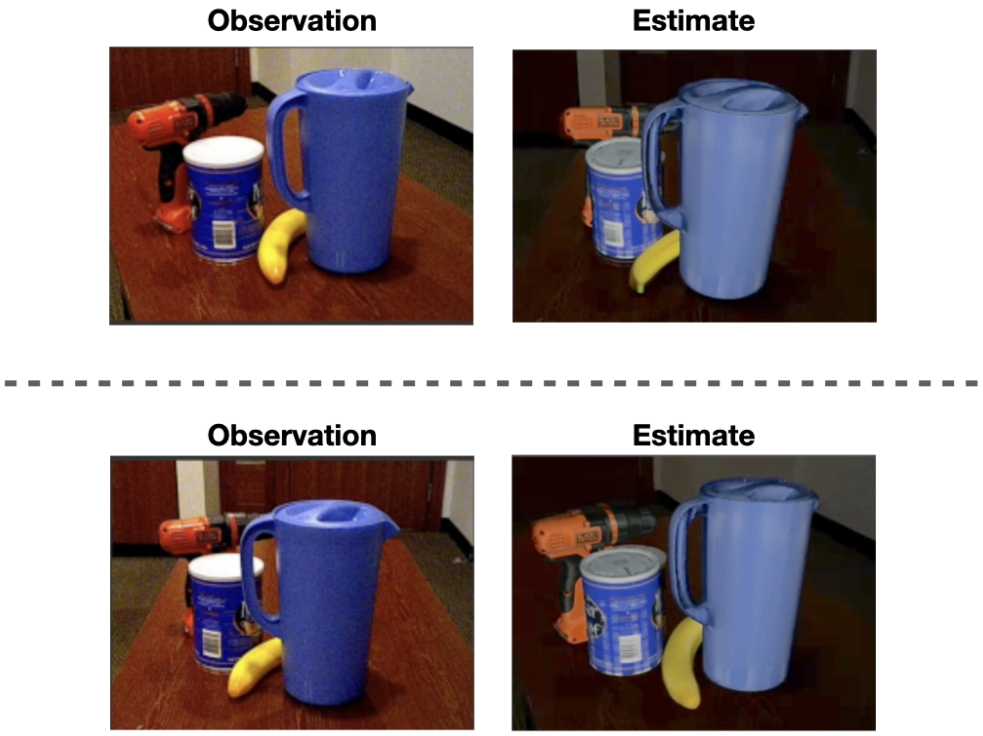


Figure 1.3: Observation and estimate at an initial viewpoint (top) when the partial observability and symmetry of the banana cause an inversion of the orientation in the pose estimation. Sometime later (bottom), he banana is viewable enough to disambiguate similar poses and correct the orientation.

deviates from the typical assumption—leveraging information and estimates from previous time-steps will help a robot maintain consistent and accurate estimates. As we will discuss further, we propose an alternative method for reasoning about the uncertainty of the pose at each frame, such that the system is more capable of making such a necessary correction.

1.1.1 Uncertainty and Inference

Particle Filters, as a form of Sequential Monte Carlo, is one such method to reason under uncertainty through probabilistic Bayesian inference [7]. While Gaussian filters [8] are constrained to only solve linear systems, particle filters are nonparametric and able to represent nonlinear systems. The distributions they can represent are not constrained to be Gaussian, which is particularly advantageous

when there are multiple reasonable hypotheses that should be considered in the belief distribution, e.g. observing rotations of an object with symmetry or representing the configuration of an over-actuated robot. Even when the sequential information is represented as a complex graphical model [9], it can be implemented as a sampling-based method through nonparametric belief propagation [10].

The current era of reasoning under uncertainty features blends of methods that have traditionally been siloed. Observation models of nonparametric inference that previously relied on expert-engineered features [11, 12] have been expanded to include data-driven models [13, 14, 15]. Differentiable filtering pipelines have enabled action models and pairwise potentials to even be learned as opposed to created based on domain-specific knowledge [16, 17, 6]. Conversely, probabilistic reasoning remains a more stable and principled underlying framework to handle the noisy estimates from neural networks. Learned methods can measure uncertainty based on the consistency of predictions when there is stochasticity added to the network, such as training the weights to be represented by a probabilistic density function [18, 19] or randomly dropping out portions of the network [20, 21].

1.1.2 Particle Deprivation

Though these methods alleviate some of the hand-tuning traditional filters and overconfidence of neural networks, the nondifferentiable particle resampling step often propels the inference process into failure modes at the time of inference. Because its posterior distribution is determined by sampling from the prior distribution, states unrepresented in the previous instance of time will continue to be unrepresented unless their distance to the particle set is within the variance of the action model. The omission of the true state within the particle set will cause the filter to maintain an incorrect estimate, even if its likelihood function is perfect.

This problem is referred to as *particle deprivation*, and it is exacerbated in robotics

due to its computational and time constraints limiting the particle set size in relation to the state space. Because of this disparity, many robotics works (particularly in the domain of mobile robot localization) have examined how to overcome particle deprivation. They examine increasing the size of the sample set and redistributing the samples as needed, known as *adaptive particle reinvigoration* [22]. In these cases, the need to recover regions of the belief through particle reinvigoration is typically signaled by heuristics centered on the likelihood function. Measuring uncertainty is indirectly inferred by examining the likelihood because these quantities are dependent on one another in Bayesian reasoning. They can be categorized as *zero-sum*, meaning any increase in one quantity directly implies a decrease in the other quantity.

It is with this emphasis on the mechanics of the reasoning about belief and doubt within a filter that this dissertation begins to deviate from previous works. Given that certain viewpoints do not contain enough visual information to disambiguate between the correct pose and other plausible hypothetical poses, it may be difficult to precisely quantify the belief and doubt associated with a pose. Some of the uncertainty surrounding a given pose might not be caused by the presence of unsupporting evidence but rather a lack of evidence. Allowing for this third quantity of ambiguity to be considered alongside belief and doubt removes the dependency between likelihood and uncertainty. While perhaps creating a more useful representation of confidence, the greater intricacy requires additional measurements beyond the standard likelihood function.

We propose additional quantification to more completely represent the confidence associated with a pose estimate. As shown in Figure 3.1, we illustrate how different observations and their respective hypothetical poses would induce varying delineations between belief, ambiguity, and doubt. In this dissertation, we introduce the counter-hypothetical likelihood function to quantify the doubt associated with a given pose. A pose estimate that can be clearly confirmed by the observation results in a relatively

high likelihood score, while a pose estimate that the observation unquestionably discounts would in turn have a high counter-hypothetical likelihood weighting. When little of the possible evidence can be definitively categorized as either supporting or unsupporting of a given pose, this implies the presence of a higher amount of ambiguity.

We validate this extension by comparing it against prior methods in a context that is more realistic for robotics applications. For the particle filter, we re-evaluate a state-of-the-art rigid object tracking framework [14] on a standard benchmark [23]. In our experiment, we disable any initialization with or usage of ground truth since it is typically not available at the time of inference. By forcing the tracking framework to recover the pose without a ground truth prior, it is more prone to mode collapse of the true pose. Our counter-hypothetical likelihood function seeks to address this failure by predicting when the samples need to be redistributed to recover pose.

We then demonstrate that higher-dimensional inference methods [6] are susceptible to particle deprivation when estimating the pose of highly occluded objects. To complete this aim, we observed there was a lack of robot pose datasets featuring heavy occlusions in the real world, as described further in Section 4.3. This gap motivated our collection and annotation of tracking a humanoid robot under heavy occlusion. By focusing on the challenge caused by limited observability of the object, we motivate the benefit of our measuring ambiguity and doubt independently of likelihood within belief propagation.

1.1.3 Thesis Statement

In this dissertation, we posit Evidential Theory [24] can improve the reliability of nonparametric probabilistic inference. Evidential Theory models the likelihood of a hypothesis, the doubt associated with it, as well as the ambiguity or ignorance of

the observation. It is a generalization of Bayesian reasoning, as Bayesian reasoning only accounts for likelihood and the resulting belief. This additional modeling allows for a separate measurement of doubt, because we no longer assume it is strictly dependent on the estimated likelihood. Synergistically, an emerging trend in deep learning research is to quantify the network's confidence in its performance, as opposed to mere confidence in a given hypothesis. Deep evidential reasoning is the learning community's response to limitations with the lack of transparency and trustworthiness in neural networks, but this representation has yet to be sufficiently incorporated within probabilistic robotics. The insights of this dissertation are our work at the intersection of the state of the art of probabilistic reasoning under uncertainty and emerging trends in deep uncertainty quantification.

In this dissertation, we propose using deep uncertainty quantification to improve the quality of the perception framework itself and incorporate it into the formalization of uncertainty in Bayesian nonparametric inference. Our proposed thesis relaxes the Bayesian assumption of nonparametric inference to allow the integration of evidential reasoning. We argue that deep evidential reasoning can be integrated into Bayesian nonparametric inference as a signal to increase the reliability of the inference.

1.2 Statement of Dissertation Scope

To address the problems of reasoning under uncertainty for applications in mobile manipulation robots, this thesis introduces integrating evidential reasoning within Sequential Monte Carlo. We present novel methods to translate the doubt explicitly modeled in evidential reasoning to redistribute its samples. To overcome particle deprivation and mode collapse, our work also demonstrates how the ambiguity of evidential reasoning can inform the balance between information visually observed and information deducted from known geometry.

More specifically, this dissertation makes the following contributions:

1. **Counter-Hypothetical Particle Filters for Single Object Pose Tracking** (Chapter 3) [25]: In this chapter, we estimate the necessary reinvigoration rate at each time step by introducing a *counter-hypothetical* likelihood function, which is used alongside the standard likelihood. The addition of our counter-hypothetical likelihood function assigns a level of doubt to each particle, allowing us to estimate doubt independently of ambiguity or likelihood. The competing cumulative values of confidence and doubt across the particle set are used to estimate the level of failure within the filter in order to determine the portion of particles to be reinvigorated. We demonstrate the effectiveness of our method on the rigid body object six-degree-of-freedom pose tracking task.

2. **The Progress LUMBER Dataset** (Chapter 4): We present the Progress LUMBER Dataset(Looking Upon a Moving BipEdal Robot), which captures 100 sequences of a walking bipedal humanoid robot. Its presence of heavy occlusions due to external obstacles separates it from previously collected robot pose datasets. The dataset allows us to test our tracking methods on a moving highly articulated object of a known model in realistic and occluded scenes.

3. **WAGER-DNBP: Weighted And Graphical Evidential Reasoning for Differentiable Nonparametric Belief Propagation** (Chapter 5): Tracking highly articulated robots poses a significant challenge due to the intricate state space and potential partial observability. We introduce deep evidential reasoning to nonparametric belief propagation to better handle noisy and error-prone tracking scenarios. This work continues the use of the counter-hypothetical likelihood to determine the reinvigoration needed at each node of the filter. However, since belief propagation allows for reinvigorating samples near neighboring nodes, we present adaptive particle reinvigoration for two random distributions. We additionally use our independent quantifications of likelihood and doubt to

estimate the ambiguity associated with an observation and hypothesis. We then use these ambiguity scores to modulate the effect each observation has on a sample's final importance weighting.

Chapter 2

Related Work

Our work examines methods to improve the robustness of robot perception for enhancing dexterous mobile manipulation. A robot's understanding of its environment hinges on both sensing low-level information about the scene accurately as well as intelligently reasoning about these extracted details over different viewpoints and timesteps. With this schema, inaccuracies or ambiguities in the initial perception can be filtered out, and the system can additionally reason about higher-level and semantic properties of the environment.

This thesis seeks to improve the performance of autonomous perception tasks by incorporating advancements in the interpretability of deep learning, namely deep evidential regression, into the underlying probabilistic reasoning system of many robotics platforms. To better explain the novelty and significance of our contributions, we cover the related work within Computer Vision for Mobile Manipulation (Section 2.1), Sequential Monte Carlo (Section 2.2), and Deep Uncertainty Quantification (Section 2.3).

2.1 Computer Vision for Mobile Manipulation

Our proposed methods are demonstrated on multiple perception applications necessary for a robot to move about and grasp objects in a real-world environment. As such, we highlight some of the tasks of the perception pipeline, as well as briefly cover historical trends for implementing them. The field has generally navigated

from hand-engineered visual features crafted by humans to latent models trained in a data-driven manner.

Though deep learning has improved the accuracy of computer vision tasks, it comes with a specific set of concerns. It learns a "black box" model to infer about new data, so engineers cannot precisely understand the relationship the model has assumed of the real world. Additionally, if errors remain, retraining the model on more balanced and comprehensive data is not an exact science, and no guarantees can be made about the hopeful improvement in performance. Not only is the latent representation opaque, but it is unclear how much an algorithm should rely on the output of a neural network. These networks can be over-confident, leaving little indication to a downstream autonomous system when the perception is critically failing.

2.1.1 Object Detection

Object detection is a crucial aspect of mobile manipulation and has been a significant challenge for computer vision in the real world for several decades. This problem space can generally be divided into inferring if a specific instance of an object is present or looking for all possible instances of a generic class [26]—a more practical use case in robotics. The Viola-Jones detectors were the first face detection network that achieved fast and accurate face detection without constraints [27]. This was accomplished by exhaustively applying sliding windows over the image. Over time, an improved feature descriptor, the Histogram of Oriented Gradients (HOG), was developed [28]. This descriptor demonstrated better generalizability to object variance in scale and shape, and it was integrated into the Deformable Parts-based Model [29, 30]. This model identified smaller components of the object and modeled object identification as a composition of these lower-level detections in post-processing.

Then data-driven neural networks began to replace engineer-designed feature de-

scriptors, significantly improving object detection performance [31]. Neural networks expanded the capabilities of generic object detection by generating category-independent proposal regions [32] instead of relying on provided bounding boxes. However, these proposal regions were still classified via category-specific features. However, the dramatic performance improvement inspired later frameworks to extend the technique by additionally classifying the proposed regions by training in an end-to-end fashion [33, 34, 35] with the most notable networks being Fast RCNN [36] and Faster RCNN [37]. As the networks deepened in complexity, they grew to additionally estimate the instance's segmentation [38]. Validation for accuracy improvements in data-hungry neural networks has motivated the need for benchmarked datasets [39, 40].

2.1.2 Object Pose Estimation

Once an accurate bounding box or region of interest of an object is generated, the pose can be estimated. These methods are categorized as either instance-level, where an exact model of the object is known, or category-level, where it is not. Category-level pose estimation has produced a large body of work [41, 42, 43, 44, 45, 46, 47], with foundational papers already having been expanded upon for greater advancements. However, since this dissertation will focus on instance-level pose estimation and tracking, we have only referred to a few sampled papers from category-level pose estimations impressive works.

Instance-level object pose estimation follows a similar history to object detection in that it originally began with hand-crafted feature-based methods. For example, when registering an object within a point cloud, known geometries between two points in an object could be encoded in Point-Pair features [48, 49]. By using the descriptor to match the corresponding 3D points between the point cloud and observed depth, the pose of the known object could recovered based on the necessary transformation for registration. Additional methods [50, 51] then focused on visual landmarks in

RGB images of an object and then used the Perspective-n-Point algorithm (PnP) to recover 3D pose from 2D features [52, 53, 28, 54]. Another traditional method, template matching [55, 56], preprocessed the features in an image of the object with a known pose. By similarly extracting the features of the captured image, it can be determined which pose in the data of previously labeled images most resembles the captured image. This match of best fit is used to infer the object's pose.

The features used in these methods struggle to have consistent descriptions of a 3D point on an object across lighting changes and occlusions. As such, the matching algorithms can fail, resulting in poor performance in pose estimation in clutter. These methods also lacked scalability, as the individual features had to be tediously engineered and tested across different objects and scenes. Because of this burden, deep learning has also been used to create data-driven features [57, 58, 59, 60]. Though learning features have become scalable, PnP and template matching can have a high computational cost during inference, leading to the exploration of regression-based deep learning pose estimation methods.

The foundational paper for pose regression for a 6D object pose estimation is generally considered PoseCNN [23]. This work initially predicted the pose using RGB information, then refined with point cloud registration using depth information. The intertwinement of this task with object detection is showcased in multiple pose estimation networks being implemented as an additional branch of an object detection network, with Deep-6DPose [61] building off of Mask R-CNN [38] and YOLO-6D [62] as an extension of YOLO [63]. Subsequent works have focused on specific cases, such as overcoming occlusion [64, 65], the lack of a mesh model [66], or transparency in the object [67, 68, 69]. Iterative methods estimate the needed transformation between an initial pose estimate and a viewed object [70, 71]. This concept has also been used to create a tracking network in which the action model is replaced by the refinement model [72]. Another tracking-focused work of note

is Pose-RBPF, which implements a Rao-Blackwellized implementation through the orientation of the pose being decoupled from translation and recovered from template matching on learned embeddings [14]

Though these methods report high accuracy on rigid objects, transferring their success to high-dimensional objects remains an open problem. Some works have looked at end-to-end regression on highly articulated objects [73]. It can be difficult to thoroughly represent possible vantage points for a rigid object in a balanced manner, but to additionally represent the configuration space dramatically exacerbates this problem. As such, multiple works have examined using deep learning for local pose information that is integrated through a factor graph backbone. One such work by Pavlasek et al. [13] presented a generative-discriminative framework for parts-based instance-level pose estimation of articulated objects, and a later work focused on this task at a category level [74]. Since a graphical model requires the tedious creation of many unary and pairwise likelihood functions, Opipari et al. presented a differentiable belief propagation network trained end-to-end [6].

This dissertation focuses on improving the performance of these Bayesian pose tracking methods while observing objects with occlusion. For a 6D pose tracking framework [14], we demonstrate the ADD and ADD-s error metrics for the pose estimates significantly increase when the object is partially observable. We then contribute a high dimensional, heavily occluded tracking dataset of a humanoid to examine the failure modes of differentiable belief propagation [6] when obstacles are present. We show that with these partial observabilities, differentiable belief propagation has increased error in its estimations of joint positions for the robot. We then show how incorporating evidential reasoning into the underlying Bayesian inference can help alleviate these failure modes.

2.2 Sequential Monte Carlo

Robotics applications must be mindful of errors in their perception system to avoid hazards when planning based on this information. Sequential Monte Carlo is a probabilistic foundation traditionally used to compensate for noise and ambiguities in local estimations. Recent works seek to harness the discriminative power of deep learning by integrating it into the filter.

Sequential Monte Carlo is still valued due to the *reliability* and *diagnosability* it provides. While inference from a neural network can be noisy, filters can be explicitly coded to provide consistent and plausible estimates. Additionally, their extracted information is modular and grounded in our mathematical models of the real world. This allows humans to better predict when they are performing poorly, as well as modify the algorithm to avoid any preventable repeats of failure. Our work combines modern techniques for associating uncertainty with a possible hypothesis for state-of-the-art pose tracking and particle reinvigoration. For this reason, we will focus on Sequential Monte Carlo.

Known by many names, Sequential Monte Carlo was conceptualized by multiple independent works. The *bootstrap filter* was first introduced as a solution to represent the nonlinearity present in tracking problems [75]. It was also presented as the Condensation algorithm by the computer vision community for visual hand tracking [4]. The mathematics community coined it the *Monte Carlo filter* [76]. As a canonical method in robotics, Dellaert et al. introduced the *particle filter* [77] for mobile robot localization from laser rangefinding inferred by Sequential Monte Carlo inference. The contribution from these methods provided a general method for nonlinear and non-Gaussian estimation, which has long been a gold standard for robotics.

Sequential Monte Carlo was also shown to integrate well within inference methods

expressed as graphical models, such as algorithms for belief propagation. Belief propagation was first presented as a probabilistic algorithm for performing exact inference on tree structures [9]. It was later extended to approximate inference on structures containing loops [78]. Isard et al. [79] implemented the connection between Sequential Monte Carlo and probabilistic graphical inference by presenting PAMPAS–PArticle Message PASsing for computer vision applications. Tracking parts-based models, Sudderth et al. applied nonparametric belief propagation to tracking [10] and, more specifically, hand-tracking [11].

2.2.1 Differentiable Filters

Even with a growing focus on neural networks, Sequential Monte Carlo has remained a useful probabilistic backbone with module models that can be trained through backpropagation. The Differentiable Particle filter was presented by Jonschkowski et al. [16], which introduced differentiable observation and action models. Instead of training the models on manufactured accuracy measurements or intermediate steps of the pipeline, they could all be trained end-to-end on the downstream task of state estimation. Similarly, Karkus et al. presented the Particle Filter Network [17], which emphasized visual understanding more.

It is important to note that though these networks can be trained end-to-end, the resampling stage of Sequential Monte Carlo is not differentiable. This has been circumvented by training up until the resampling step [16] or parametrizing the resampling step for a differentiable approximation (soft resampling) [17]. It's also been modeled as a deterministic and differentiable optimal transport problem [80], which can be solved via the Sinkhorn algorithm [81], by Corenflos et al. [82].

2.2.2 Resampling

We emphasize the resampling step because resampling's ability to disregard the samples nearest the true state has been a well-studied portion of the particle filter algorithm. One suggestion has been to anneal the likelihood function such that its shape oscillates between a uniform distribution and the true form of the distribution [83]. This process prevents particles from being stuck in the local optima of nonlinear likelihood functions. This issue can also be partially alleviated by having more samples to distribute. KLD sampling [22] monitors the distribution of the particles by discretizing the state space and tracking the occupancy. It then calculates the necessary number of samples to theoretically represent the true posterior distribution through the Kullback-Leibler divergence to augment the particle set. Robotics applications often leverage Sequential Monte Carlo due to the nonlinear problems of their domain, but computation time constraints often prevent them from adding more samples.

For this reason, many robotics works have looked at *redistributing* a fixed number of particles as needed. Lenser et al. proposed that the unnormalized values of the importance weightings could indicate when the particle filter was in failure mode [84]. Since likelihood functions were hand-designed at the time, the engineer should have an approximation for a typical unnormalized weighting of the true state. As the average likelihood of the set dropped below the threshold, a higher portion of the samples would be reinvigorated by being initialized from the proposal distribution. Since the optimal weighting can vary between trials, Augmented Monte Carlo [85, 86] was proposed to use decaying rates to track if the short-term average of the weights had fallen below the long-term average of the weights to signal when the filter had lost track of the true state. Unfortunately, particle deprivation is most often caused by poor initialization, which would not be identified in this proposed modification. Recent work [87] has examined how a neural network's noisy output

can be fused with sampling from the prior distribution to create a more accurate posterior distribution.

2.2.3 Evidential Reasoning for Sequential Monte Carlo

Bayesian inference has only quantified the evidence supporting a hypothesis through the likelihood function. We augment differentiable filters to additionally learn to quantify the ambiguity and doubt associated with a given hypothesis and observation. After we estimate doubt independently of likelihood, we use these quantities to inform the portion of samples that should be initialized during resampling. The measurements of ambiguity associated with an observation convey the usefulness of the visual information at each graph node. Based on this score, we extend nonparametric belief propagation to learn to modulate the influence each observation has on a given hypothesis's final importance weighting.

2.3 Deep Uncertainty Quantification

Though neural networks have demonstrated high accuracy in performance overall, they can be overconfident even when inaccurate. In some of these problem domains, the noisy output of neural networks can induce discrimination [88] and even danger [89]. While many robotics applications have countered this with more reliable underlying reasoning systems [90, 2], there has also been a demand for *explainable artificial intelligence* [91] to improve the interpretability of neural networks themselves. This vision outlines a sub-goal for users to "appropriately trust" artificial intelligence systems and provide them with an accurate mental model of when it will succeed and fail.

The deep learning community has avoided regressing to hard-coded and hand-designed algorithms for reliability and interpretability. In this vein, many works have investigated methods to indicate when a given estimate from a neural network cannot

be trusted through *deep uncertainty quantification*. Note these techniques are not the same as confidences from the network on the original classification or regression task on a single data example [21]. Deep uncertainty quantification examines new supervisory signals and frameworks for a neural network that can be more indicative of the network's performance on unseen data. Additionally, neural networks are extremely unreliable when tested on data outside the training data domain, so out-of-distribution detection has been an essential form of model uncertainty quantification for deep learning. There has been promise in classifying the category of the domain shift [92] and their implications to safety [93].

2.3.1 Uncertainty in Deep Learning

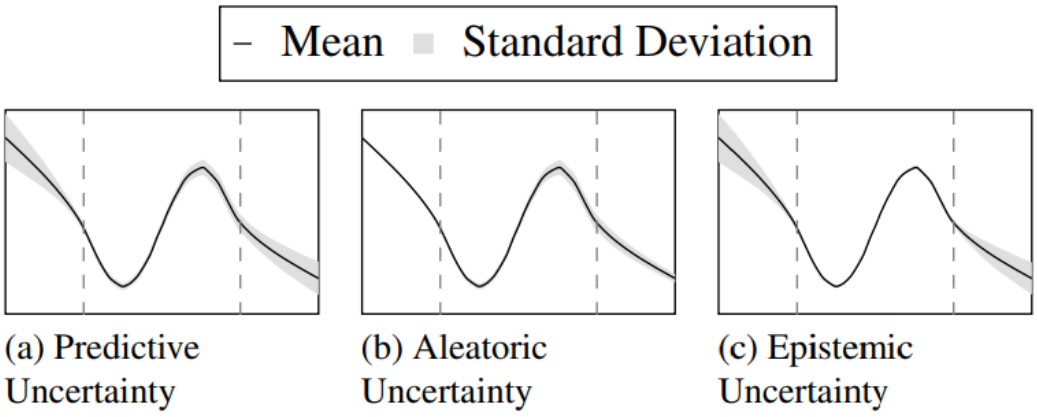


Figure 2.1: Visualization of categories of uncertainty. The dotted lines represent the bounds of the training data shown in the model. a) Predictive uncertainty is the total uncertainty comprised of aleatoric and epistemic uncertainty. b) Aleatoric uncertainty stems from unpredictable randomness in the data and cannot be reduced by more training data. c) Epistemic uncertainty is caused by the model's lack of proper knowledge and occurs when tested on data outside of its training distribution. from [3]

Predictive uncertainty can be separated into two categories—aleatoric, caused by the inherent randomness of the data, or epistemic, caused by a lack of knowledge of the model [94]. Epistemic uncertainty arises from uncertainty about the model itself,

such as the network's architecture or weights. It also applies to out-of-distribution, in which the training data distribution is dissimilar enough to the testing example that the previously seen examples are not helpful. Epistemic uncertainty cannot be reduced by adding more examples of more relevant data. On the other hand, aleatoric uncertainty cannot be reduced by more data. It refers to the randomness and unpredictability of the data. This applies to errors or noise in the sensor or partial observability.

A depiction of the variance of both uncertainties concerning the training data domain is shown in Figure 2.1. In essence, aleatoric uncertainty is meant to show more variance when there is randomness or noise associated with the data, and epistemic uncertainty is meant to increase uncertainty as the data extends past the bounds of the previously seen data. Many works have analyzed inference methods on their ability to quantify both of these sources of uncertainty individually [95, 96, 3, 97].

2.3.2 Bayesian Methods

Bayesian Neural Networks (BNNs) [98, 99, 100] further quantify uncertainty by making the neural network stochastic. The weights are modeled as a prior Gaussian distribution, with training examples modifying the mean and variance of each weight distribution. At the time of inference, the network determines the weights of its model by sampling from this distribution. This stochasticity allows the framework to have slightly different estimates from the same architecture for the same input, such that the repeatability and agreement between these outputs can be analyzed to determine the degree of uncertainty.

Another method of stochastic uncertainty quantification is achieved by randomly changing the architecture of the network. Dropout [20] is a method that was originally proposed to prevent overfitting in the network. Activation layers of the

network and their corresponding incoming and outgoing connections are randomly chosen to be temporarily removed at different iterations of training. Monte Carlo Dropout (MC-Dropout) by Gal and Ghahramani [21] showed that testing the data through a feedforward version of the network with dropout multiple times can quantify prediction uncertainty. This method is thought to be less computationally draining than BNNs, as it has similar implementations for training and testing.

2.3.3 Ensemble Methods

Determining predictive uncertainty through analyzing multiple estimates was continued with the work of deep ensembles [101]. As opposed to adding stochasticity to a single network and measuring the repeatability of output, this methodology allows for different networks and training data entirely when quantifying consensus. With the plug-and-play nature of pre-trained neural networks, it has gained increasing popularity. For example, in robotics, a camera on a mobile manipulator measured the alignment of pose estimates from several networks at each frame. The frame that produced the most similar estimates across the networks was chosen for viewpoint selection to determine the object's pose to grasp [102].

2.3.4 Deterministic Single-Network Methods

The previously suggested methods all require multiple passes of a network at the time of inference for uncertainty quantification, which is not always feasible for robotics applications. The final category of uncertainty quantification seeks to use a single network with a single set of weights on a single inference pass to estimate the uncertainty. Given our motivation to improve performance when time and resources for the estimation are limited, we draw inspiration from this category.

Several methods implement this network as an external neural network for the prediction. The introduction of a separate and external network to quantify the

uncertainty of the prediction framework was introduced to prevent bias of any bias in the uncertainty quantification [103]. This idea was built upon by having the secondary network not only reason about the estimate but also consider how the current input's representation compares to the distribution of representations from the training data to inform its uncertainty quantification [104]. Another example is applied to the classification task, which sums the probabilities of all classes. It demonstrates how lower sums indicate the testing data is out of the distribution of the training data [105].

2.3.5 Deep Evidential Reasoning

The idea of quantifying the ignorance of the model itself stems from Evidential Theory, or Dempster-Shafer Theory [24, 106]. This work was originally applied to discrete probabilities and formalized how varying and possibly contradictory sources of information could be unified for probabilities on what one might conclude. It is important to note that pushback on this ideology has occurred due to concerns it is erroneously used to quantify *probability* as opposed to the more appropriate *probability of provability* [107]. Evidential Theory models belief in an outcome, the disbelief of an outcome, and ambiguity or ignorance in measuring the possibility of an outcome. These quantities are also combined to give the notions of plausibility and implausibility, as visualized in Figure 2.2.

Evidential Theory has been the foundation for another wave of deep uncertainty quantification—deep evidential reasoning [108, 109, 110]. Whereas BNNs place a prior distribution on the network weights, deep evidential reasoning places a prior on the likelihood function itself. This was originally applied in the domain of discrete classification [109, 110, 108]. An example is adding a semantic class of 'unknown' to the output set, where the network can assign some probability to avoid penalization from erroneously assigning that quantity to a real class during softmax.

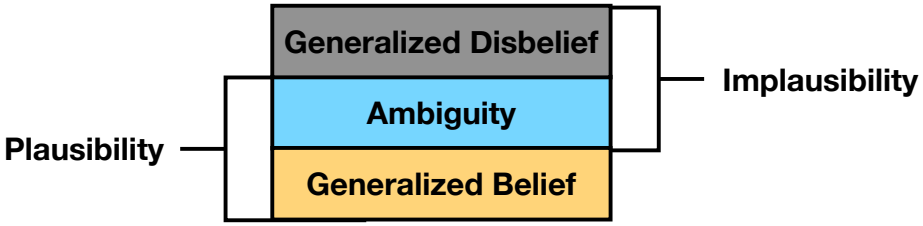


Figure 2.2: Evidential Theory quantifies the evidence of support (generalized belief) and unsupporting evidence (generalized disbelief) of an outcome, and it additionally quantifies the ambiguity or ignorance associated with the deduction itself. These quantities are used to measure the plausibility and implausibility of an outcome.

Amini et al. extended deep evidential reasoning to the continuous problem space [111] and coined the term *deep evidential regression*. They train a network to predict the associated evidence associated with an estimate. Their likelihood function is a Gaussian likelihood function, and the network learns to predict the parameters of this distribution. This method did not require out-of-distribution training data or sampling at the inference time to measure the predictive uncertainty.

Similarly, our work uses a single and deterministic network to quantify uncertainty for a regression task. We also build off the Evidential Theory paradigm to distinguish between ambiguity and doubt. Our work is inspired by a loss term in Deep Evidential Regression that seeks to penalize learning evidence for incorrect estimates. However, instead of quantifying uncertainty, our work aims to explicitly quantify doubt independently of ambiguity. Additionally, this dissertation focuses on integrating the estimated uncertainty measurements into a nonparametric filter.

Chapter 3

Counter-Hypothetical Particle Filters for Single Object Pose Tracking

Particle filtering is a common technique for six degrees of freedom (6D) pose estimation due to its ability to tractably represent belief over object pose. However, the particle filter is prone to particle deprivation due to the high-dimensional nature of the 6D pose. When particle deprivation occurs, it can cause mode collapse of the underlying belief distribution during importance sampling. If the region surrounding the true state suffers from mode collapse, it is challenging to recover belief since the area is no longer represented in the probability mass formed by the particles. This failure mode is depicted in the previously presented Figure 1.3, where the initial occluded estimate results in an incorrect pose of the banana. The rendered pose of this estimate aligns well with the border of the visible banana, but we can see a 180° flip of its orientation. Sampling off of the previous set of samples would not correct the pose because the variance of the action model cannot represent such a large area of the state space. Instead, determining a portion of the samples to be reinitialized through particle reinvigoration allows the filter to recover the true pose. Previous methods mitigate this problem by randomizing and resetting particles in the belief distribution, but determining the frequency of reinvigoration has relied on hand-tuning abstract heuristics.

In this chapter, we estimate the necessary reinvigoration rate at each time step by introducing a *counter-hypothetical* likelihood function, which is used alongside the

standard likelihood. Inspired by the notions of plausibility and implausibility from Evidential Reasoning, the addition of our counter-hypothetical likelihood function assigns a level of doubt to each particle. The competing cumulative values of confidence and doubt across the particle set are used to estimate the level of failure within the filter and to determine the portion of particles to be reinvigorated. We demonstrate the effectiveness of our method on the rigid body object 6D pose tracking task. As previously shown in Figure 1.3, our method aims to identify when a larger error in the estimate is due to mode collapse and correct the pose by redistributing the samples. This work has led to the publication, 'counter-hypothethical Particle Filter for Single Object Pose Tracking' [112].

3.1 Introduction

As robot assistants become tasked with accomplishing complex chores, such as preparing a meal or tidying a room, they must be able to interact with various objects. Object pose estimation in unstructured scenes remains challenging due to the ambiguity in perception, which arises from occlusion and symmetries. Particle-based inference methods have been widely applied to six degree of freedom (6D) object pose estimation and tracking due to their ability to represent high-dimensional spaces with finite sample sets [113, 114, 14]. These methods model estimation uncertainty and can maintain multiple possible pose hypotheses, which provides robustness in challenging scenarios such as object occlusion and ambiguous symmetries.

Despite these promising properties, particle filter algorithms are forced to limit the size of their sample sets to ensure tractability for robotics applications. When applied to 6D pose tracking, particle filters typically can afford only a small number of samples compared to the overall size of the continuous state space. Since the sample set cannot completely cover the space, certain regions of the state space will contain no particles, making their representation in the belief distribution collapse. This

phenomenon is called *particle deprivation* and can occur due to poor initialization or the stochasticity of importance sampling. Regaining belief in these regions is challenging and can cause the filter to converge to an incorrect local optimum.

One strategy for mitigating particle deprivation is *particle reinvigoration* [86], in which reinitialized samples are routinely added to the set. However, determining the portion of particles to be reinvigorated at a given iteration often requires tedious hand-tuning through trial and error. Many adaptive approaches have been proposed to mitigate this challenge by leveraging the information in the likelihood function [115, 84, 116]. The likelihood function gives an importance weighting to each sample by measuring the correspondence of the hypothesis to the observed sensor data. However, it only provides a relative weighting of which samples are *better* or *worse*, and no indication of the absolute error in the sample set.

We introduce the counter-hypothetical particle filter (CH-PF) to counteract particle deprivation in high-dimensional state spaces. Our method proposes quantifying the confidence that the true state is unrepresented in the sample set. To this aim, we model the evidence *against* a particular hypothesis, termed the *counter-hypothetical* likelihood. Our work measures this weighting independently of the traditional likelihood through the lens of Evidential Reasoning (Dempster-Shafer Theory) [106]. This framework argues that the plausibility and implausibility of proving an outcome can be based on different factors and are not zero-sum due to the potential overlap and ambiguity of the underlying evidence. Each particle is given both a likelihood and a counter-hypothetical likelihood weight. Our method utilizes both likelihoods to quantify the cumulative confidence and doubt across the sample set. The relationship between these values is used to reason about the likelihood that the true state is underrepresented in our sample set and, in turn, used to compute an adaptive rate of particle reinvigoration.

We propose the counter-hypothetical particle filter, a particle filtering algorithm

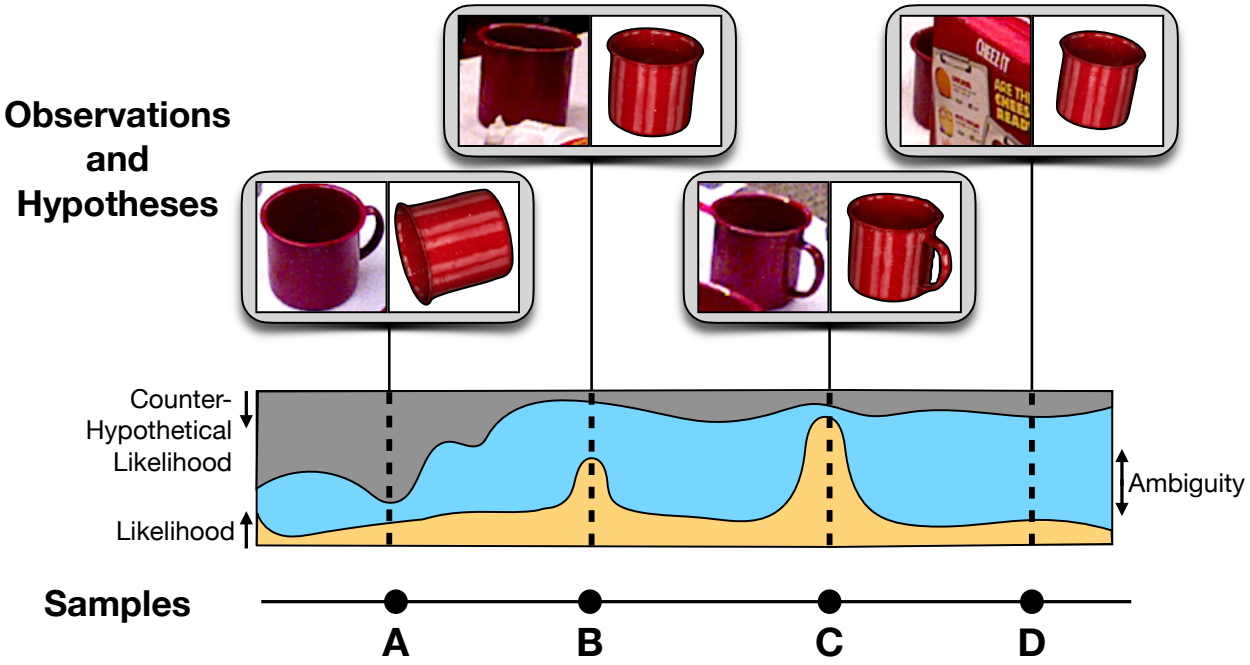


Figure 3.1: We quantify both the evidence against a given estimate (gray), as well as in support of it (yellow). We estimate these quantities independently of one another because they are not zero-sum due to ambiguity in the observation (blue). The relative magnitudes of these weightings fluctuate based on the quality of both the observations and estimates, as illustrated by a mug that is (A) unambiguously unlikely (B) plausible yet ambiguous due to the occluded handle (C) unambiguously likely (D) highly ambiguous.

designed to mitigate particle deprivation for 6D pose tracking in challenging environments. We introduce a counter-hypothetical likelihood and explain how its utilization alongside the traditional likelihood counteracts particle deprivation by leveraging implausibility information, as illustrated in Figure 3.1. We evaluate CF-PF on 6D single object pose tracking on the YCB Video Dataset [23]. Our method achieves better accuracy for cases of high occlusion, particularly when depth data is unavailable.

3.2 Related Work

3.2.1 Object Pose Estimation and Tracking for Robotics

Pose estimation and tracking have received considerable attention in the robotics community. In recent years, various works have demonstrated the capability of data-driven methods to provide discriminative pose estimates over a single view [23, 61, 117, 118], or pose tracking over a sequence of observations [72]. These methods have achieved impressive results but are prone to inaccuracies, particularly in challenging scenes with heavy clutter. Probabilistic inference methods instead maintain belief over pose estimates to provide additional robustness for robotic manipulation applications [119]. We focus on probabilistic inference for object pose estimation and tracking, specifically on particle filtering.

Particle filtering is an iterative inference algorithm that can represent an arbitrary nonparametric belief distribution using a set of weighted particles sampled from the state space [115]. Particle filtering is a common technique for 6D pose estimation and tracking [114, 2, 120, 121] due to its ability to approximate high-dimensional state spaces efficiently and represent multiple competing hypotheses in the belief. More recently, deep convolutional neural networks (CNNs) have been applied to particle filtering for pose estimation. Deng et al. [14] create an observation model from autoencoder embeddings for use within a Rao-Blackwellized particle filter. Recently, this method has been extended to category-level tracking [46]. Though the main contribution of our work is introducing the counter-hypothetical likelihood function, we also demonstrate how it could be similarly learned in an end-to-end fashion.

To mitigate challenges associated with fully sampling the high-dimensional 6D pose space, previous works have leveraged domain-specific knowledge such as physical constraints [122], robotic arm joint angles [123, 113], or context information [124]. Belief propagation using particle belief representations [125] has been applied to parts-

based object models to factor the high-dimensional articulated object localization task [126, 13]. Similarly, our work addresses the problem of particle deprivation in sampling-based methods for manipulation tasks. We aim to do so through adaptive reinvigoration and do not rely on provided object-environment interaction models or parts-based object models.

3.2.2 Robust Particle Filtering

Many works have focused on mitigating particle deprivation. One approach is annealing, in which the distribution of importance weights is smoothed according to a hand-tuned schedule to avoid collapsing modes during importance sampling [83]. Pfaff et al. propose an adaptive method to smooth the importance weights using local density estimation around each particle for mobile robot localization [127]. In contrast, CH-PF does not require modification to the sampling weights but instead handles particle deprivation through reinvigoration.

In the global localization stage of Monte Carlo localization for mobile robots, particle deprivation is a common problem [115], motivating many works to reinitialize samples as needed. One approach is to sample from an inverse distribution based on the sensor readings [128] or "reset" a subset of particles when the average likelihood of the sample set is low [84]. Augmented Monte Carlo Localization [86, 85] extends this idea by performing particle reinvigoration from a uniform distribution at a rate proportional to the difference between the long- and short-term averages of the particle weights instead of a fixed threshold. These methods require a sensor model from which samples can be efficiently drawn, which is challenging to model in RGB images. Fox et al. proposed modifying the size of the sample set based on the quality of the sample approximation [116]. Zhang et al. propose a self-adaptive method that maintains a fixed sample size augmented by samples from a "similar energy region" [129]. This method requires a discretization of the state space. Recent

work in localization leverages the estimates of a neural network by sampling from this proposal at a fixed rate and fuses the particles into the distribution through importance sampling [87].

Each of these methods uses the likelihood weights of the particles to estimate the quality of the sample set. CH-PF instead uses a separate source of information, the counter-hypothetical likelihood, alongside the likelihood function, to estimate the overall quality of the particle set. We draw inspiration from Evidential Reasoning [106] for measuring the evidence disproving a hypothetical estimate separately from the likelihood function that looks for supportive evidence.

3.3 Background: Particle Filtering

We consider the problem of tracking a known object over time. Given a sequence of RGB images or RGB-D data, $z_{1:t}$, we seek to localize the pose, $x_t \in \mathcal{X}$, of an object at time t. We also model any motion to the system caused by either user input or jittering with $u_{1:t}$. Here, $\mathcal{X}$ represents the space of 6D poses comprised of 3D translation and 3D rotation.

The Bayes filter seeks to model the posterior distribution of the state, $p(x_t \mid x_{1:t-1}, z_{1:t}, u_{1:t})$ by iteratively updating the distribution at each timestep t. The posterior is called the *belief* of x_t, $bel(x_t)$. At each time step, the predicted belief, $\widehat{bel}(x_t)$, is obtained by applying the action model to the prior belief distribution. Employing the Markov assumption:

$$\widehat{bel}(x_t) = \int p(x_t \mid x_{t-1}, u_t) bel(x_{t-1}) dx_t \tag{3.1}$$

We can then update this distribution based on the current observation, z_t, to

estimate the posterior distribution:

$$bel(x_t) = p(z_t \mid x_t)\widehat{bel}(x_t) \tag{3.2}$$

3.3.1 Particle Filtering

The particle filter is a Bayes filtering algorithm in which the belief distribution $bel(x_t)$ is a nonparametric distribution approximated by a particle set, $\mathbb{X}_t$:

$$\mathbb{X}_t = \{(x_t^1, \pi_t^1), (x_t^2, \pi_t^2), \ldots, (x_t^N, \pi_t^N)\} \tag{3.3}$$

Each particle, x_t^i has a corresponding weight, π_t^i. The predicted belief in Equation (3.1) is formed by applying action u_t to each particle in the previous sample set, $\mathbb{X}_{t-1}$.

The particle set, $\mathbb{X}_t$, is generated through *importance sampling*, where the target distribution is $bel(x_t)$ and the proposal distribution is $\widehat{bel}(x_t)$. Samples from the proposal are drawn with replacement, in which the probability of a particle being drawn is proportional to its weight, π_t^i. Typically, the weight is computed using a *likelihood function*, $\mathcal{L}(x_t^i)$, which represents the observation model:

$$\pi_t^i = \mathcal{L}(x_t^i) := p(z_t \mid x_t^i) \tag{3.4}$$

3.3.2 Particle Deprivation and Particle Reinvigoration

If the proposal distribution does not include samples close to the true value of the state, the probability of sampling values in this region is negligibly small. This phenomenon is known as *particle deprivation* and is illustrated in Figure 3.2 (left). It can occur due to poor initialization, unmodelled movements in the state, or a series of unfortunate draws in importance sampling, causing the particle set to

converge to local optima.

A common approach to mitigating particle deprivation is *particle reinvigoration*, in which particles are drawn jointly from the predicted belief, $\widehat{bel}(x_t)$ and a candidate distribution, $\phi_{cand}(x_t)$. Choices of candidate distribution might include a uniform distribution over the region of interest of the state space or a wide Gaussian distribution around an initial estimate. This modification allows importance sampling to draw from outside the sample set, reintroducing samples in underrepresented regions. A hyperparameter α, where $0 \leq \alpha \leq 1$, controls the proportion of samples to be drawn from $\phi_{cand}(x_t)$. The final particle set is defined as the union of particles drawn from each set:

$$\begin{aligned} \mathbb{X}_t = &\{(x_t^1, \pi_t^1), , \ldots, (x_t^{\alpha N}, \pi_t^{\alpha N})\} \\ &\bigcup \{(x_t^{\alpha N+1}, \pi_t^{\alpha N+1}), \ldots, (x_t^N, \pi_t^N)\} \end{aligned} \tag{3.5}$$

where $x_t^i \sim \phi_{cand}$ for $1 \leq i \leq \alpha N$, and $x_t^j \sim \widehat{bel}$ for $\alpha N < j \leq N$. Note that in practice, αN is constrained to be an integer.

3.4 Counter-Hypothetical Particle Filter

Selecting the reinvigoration rate, α, is challenging in practice. Sampling from the candidate distribution too frequently can discard key information from the belief distribution, while sampling from the belief distribution too frequently could lead to particle deprivation. *Adaptive particle reinvigoration* mitigates this challenge by determining the frequency with which to draw from each distribution online at each time step. The counter-hypothetical particle filter (CH-PF) adaptively selects the reinvigoration rate, α, such that it fluctuates in accordance with the portion of the true belief distribution estimated to be underrepresented by $\mathbb{X}_t$.

One method of achieving adaptive particle reinvigoration is Sensor Resetting

Localization (SRL) [84]. This method defines a probability threshold, β, representing a threshold for "good" unnormalized likelihood values. The reinvigoration rate is defined as:

$$\alpha = 1 - \left(\frac{1}{\beta N} \sum_{i=1}^{N} \mathcal{L}(x_t^i) \right) \tag{3.6}$$

CH-PF builds on this equation, using Counterhypothetical likelihood to compute the reinvigoration rate instead of a probability threshold.

3.4.1 Counter-Hypothetical Resampling

To motivate our proposed method for determining the reinvigoration rate, we first rewrite Equation (3.6):

$$\alpha = 1 - \frac{\sum_{i=1}^{N} \mathcal{L}(x_t^i)}{(\sum_{i=1}^{N} f(x_t^i)) + (\sum_{i=1}^{N} \mathcal{L}(x_t^i))} \tag{3.7}$$

where $f(x_t^i) := \beta - \mathcal{L}(x_t^i)$. With this notation, the numerator and right-hand side of the denominator are an aggregate measurement of the likelihood of the sample set. The left-hand side of the denominator, $\sum_{i=1}^{N} f(x_t^i)$, measures the poor performance across the sample set. In this way, calculating the rate of particle reinvigoration in SRL can be seen as simultaneously measuring the positive performance and poor performance of the sample set. However, the measure of poor performance, $f(x_t^i)$, is dependent on the measure of positive performance, as it is defined by $\mathcal{L}(x_t^i)$. This dependency is due to traditional Bayesian probability, in which the observed probability of a state being true and the probability of a state being false are always zero-sum.

Our method relaxes this assumption by taking inspiration from evidential reasoning, also known as Dempster-Shafer Theory [106]. This paradigm allows for the evidence discounting an event, *generalized disbelief*, to be quantified independently of the evidence associated with supporting an event, *generalized belief*. As shown in

Figure 2.2, evidential reasoning models these concepts and the presence of ambiguity or ignorance [130]. Generalized belief is a measurement of all evidence that undeniably supports an event and is bounded from above by plausibility because plausibility includes ambiguity in the belief. Similarly, generalized disbelief quantifies the evidence that works to disprove an event, while implausibility is an upper bound that considers ambiguity as well.

We posit that this framework is apt for evaluating object poses based on images. The occlusions and geometric symmetries suggest ambiguity in how a given pose is supported or unsupported by the evidence, motivating us to measure these quantities independently. We consider the likelihood function analogous to Evidential Theory's notion of generalized belief and, therefore, introduce a counter-hypothetical likelihood to function similarly to Evidential Theory's generalized disbelief.[1]

We design the counter-hypothetical likelihood to measure how the observed image provides evidence *against* the hypothetical proposed state. Our approach replaces $f(x_t^i)$ from Equation (3.6) with a counter-hypothetical likelihood, which is estimated independently of $\mathcal{L}(x_t^i)$. We introduce a function to reason about the confidence of a state counter to our given hypothesis, the *counter-hypothetical likelihood*, $\mathcal{C}(x_t)$. By comparing the quantities of the (unnormalized) likelihoods, $\mathcal{L}(x_t^i)$ and $\mathcal{C}(x_t^i)$, across the proposal distribution, we can reason about the cumulative confidence and doubt in our sample set. To this end, we redefine α, the reinvigoration ratio, as follows by modifying Equation (3.7):

$$\alpha = 1 - \frac{\sum_{i=1}^{N} \mathcal{L}(x_t^i)}{\sum_{i=1}^{N} \mathcal{C}(x_t^i) + \sum_{i=1}^{N} \mathcal{L}(x_t^i)} \tag{3.8}$$

We then sample αN particles from ϕ_{cand} in accordance with our doubt in the set,

[1] A common critique of Evidential Theory is its appliance to true probability, instead of the probability of provability [107]. We do not directly use generalized belief to reason about the true underlying probability distribution. Our methods take inspiration from Evidential Theory to measure doubt independently of confidence.

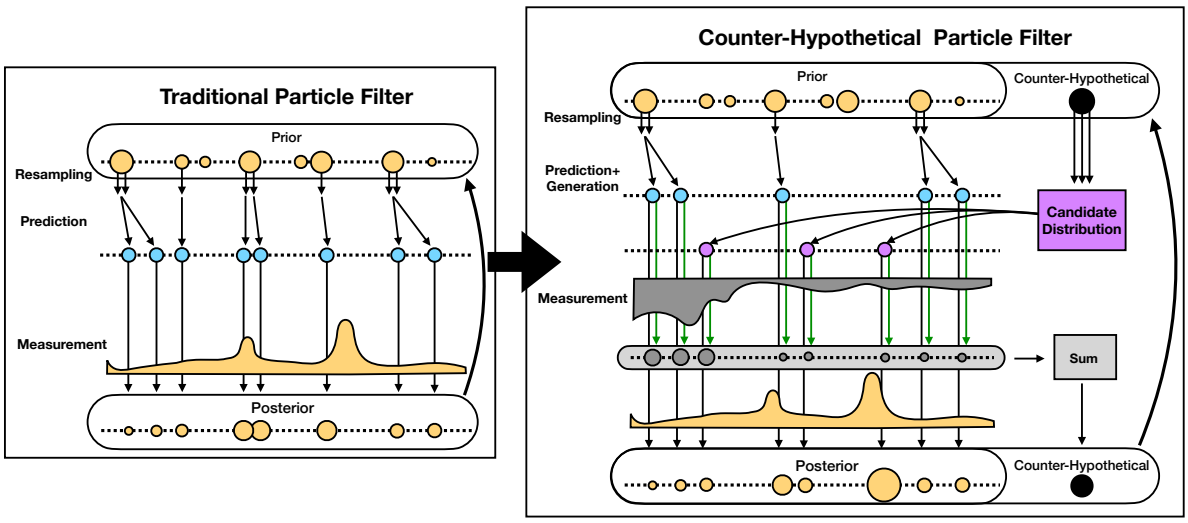


Figure 3.2: An illustration of the counter-hypothetical extension to the resampling step of the particle filter, using visuals from the Condensation Algorithm [4].

and sample the remaining $(1-\alpha)N$ particles from $\widehat{bel}$ based on our confidence in the set. As such, the counter-hypothetical likelihood quantifies our notion of generalized disbelief, which controls the amount of particle reinvigoration to be performed.

A visualization of this algorithm is presented in Figure 3.2. In the traditional particle filter (left), each iteration begins with a set of weighted particles (top). The samples are drawn with replacement, and then the action model and diffusion are applied to create the prediction distribution (blue). Each sample is passed through the likelihood function (yellow) to receive a weighting. This posterior distribution then becomes the prior for the next iteration. In our proposed modification (right), each iteration begins with a set of weighted particles (top), as well as a weighting for the counter-hypothetical (black). In the resampling stage, only five of the eight particles are created by sampling off the prior distribution (blue), and the counter-hypothetical weighting causes three samples to be randomly sampled from the candidate distribution (pink). All of these samples are passed through the counter-hypothetical likelihood to be assigned a counter-hypothetical weighting (gray). These raw weightings are summed to create a new, singular counter-hypothetical weighting representing the doubt across the set. All samples are also passed through

the standard likelihood function (yellow). The posterior distribution and counter-hypothetical weighting then become the inputs for the next iteration.

3.4.2 Counter-Hypothetical Likelihood for 6D Pose Estimation

In the context of our application, we design the counter-hypothetical likelihood to signal when the pose tracking is in a failure mode. Unlike typical likelihood functions that must be precisely crafted or trained to ensure the most accurate samples have the highest weights, the counter-hypothetical likelihood function can be more crudely or intuitively constructed.

For a simple example, consider how the captured depth data can be compared against a rendered depth image of a sample at a candidate pose. A traditional likelihood function might assign weightings based on the number of pixels of the object that have rendered and captured depths within a given threshold. This heuristic can be noisy due to the presence of occlusions and difficult to tune. On the other hand, the counter-hypothetical likelihood could measure the number of pixels in which the rendered depth is *less* than the captured depth. A potential occlusion can explain away the measured depth being significantly closer than the rendered depth, while the reverse indicates the given pose is wrong.

In this work, we use deep learning for the counter-hypothetical likelihood function and use the encoder architecture of PoseRBPF [23]. We also use their synthetic training data setup. However, instead of training the representation in an auto-encoder manner, we leverage the true pose information in synthetic data. We generate a synthetic scene containing the object and occlusions paired with a hypothetical pose. For the hypothetical pose, half the training data are positive samples, where the hypothetical pose is only slightly perturbed from the pose used to render the object. In the rest of the training data, the hypothetical pose is randomly generated to be significantly different than the rendered pose. Crops of the synthetic scenes

are the inputs to duplicates of the encoder network, the output embeddings from which are passed together through three fully connected layers. The network is trained with binary cross-entropy loss. Through this fashion, the classifier learns to estimate when a hypothetical pose is misaligned with the given observation. At test time, the scores are used as the counter-hypothetical weightings.

3.5 Experiments

To evaluate the proposed counter-hypothetical likelihood, we measure key performance metrics on the YCB-Video Dataset, a benchmarked real-world dataset [23]. We use their test set of 12 sequences, totaling $20,738$ images. We implement a standard particle filter to estimate the 6D pose of a given object across the video sequences. Our results test on both RGB and RGB-D data. All results are variants of the same particle filter, using the same likelihood function provided by PoseRBPF [14] and use the same number of particles (50). However, each baseline has a different strategy for combating particle deprivation, such as reinvigoration or Rao-Blackwellization. Whenever a candidate distribution is needed for initialization or reinvigoration, a uniform distribution of orientations located in the estimated 2D bounding box from PoseCNN [23] is used. Depth values are sampled from a uniform distribution, but when depth data is present, it is sampled off the measured depth for the object's location.

3.5.1 Baselines

We compare against methods for particle deprivation specifically designed for 6D pose estimation and adopt other techniques common in mobile robot localization.

Annealing [83] does not use any particle reinvigoration but rather has an annealed likelihood function that cyclically smooths the likelihood weightings.

SRL [84] performs adaptive particle reinvigoration from the candidate distribution

by comparing the average unnormalized likelihood weighting to a predetermined user threshold. This is a minimum cosine similarity between the embeddings in PoseRBPF.

Aug. MCL [86, 85] also performs adaptive particle reinvigoration from the candidate distribution, but the threshold is determined at each time step by user-defined decay rates.

MCL + E2E [87] has a fixed number of samples from the predicted distribution, with the remaining being sampled off a neural network estimate. We sample off a Gaussian distribution centered at the full 6D pose estimate provided by PoseCNN for the current frame.

PoseRBPF [14] is run as described in the publication, but with turning off any ground truth information used in the system. The original implementation ensures the initialization is close to the ground truth orientation or it is completely reset. In our experiment, it is only reset when the likelihood weighting of the estimate drops below a threshold. We also include its suggested variant, **PoseRBPF++**, in which half the samples are reinvigorated at each time step from the candidate distribution.

CH-PF, Annealing, SRL, Aug. MCL, and MCL + E2E filter all six dimensions to better test their ability to withstand particle deprivation. Through their Rao-Blackwellized implementation, PoseRBPF and PoseRBPF++ filter across only three dimensions of continuous state space (the location) because the orientation space is discretized.

3.6 Results

We present results with the absolute and symmetric pointwise matching errors between the estimated and ground truth pose (commonly referred to as ADD and ADD-S, respectively) [23]. When considering points from the object's mesh at an

estimated pose and the true pose, ADD is the average distance between corresponding points. ADD-S does not measure the distance between corresponding points, but rather the minimum distance to *any* point of the other pose. ADD-S is more applicable for objects with an axis of symmetry that makes a single true pose annotation difficult, such as the bowl. These errors are analyzed by viewing the Area Under the Curve (AUC) score of each method up to 10 cm error, as in other YCB works. The higher the AUC score, the lower the distance error threshold most estimates fit within. An intuitive understanding of this metric depends on the gripper used to grasp the object, as the amount of pose error the gripper can overcome varies. Full quantitative results are shown in Figure 3.3.

Our analysis shows that for variants of a particle filter using the same weights but different methods for maintaining particle diversity, there is little deviation in performance across the entire dataset. The Rao-Blackwellized implementations, PoseRBPF and PoseRBPF++, have the best performance. However, a disparity in accuracy is visible when looking at the sequences where the given object is occluded. In these cases, PoseRBPF has a decrease in performance. We hypothesize that this is due to the orientation filtering mechanism in PoseRBPF, which makes the system very sensitive to the quality of the initialization; an initial orientation that is flipped is difficult to correct, even with reinvigoration. For this reason, the poor initialization occurring in partially observable scenes lowers its accuracy ranking concerning the other methods. Aug. MCL can work well when the filter loses track of the pose, but in the case of poor initialization, it does not record a value for an ideal likelihood threshold and, therefore, performs little reinvigoration during occlusion. SRL and MCL + E2E perform on par with or without occlusions being present. Our method performs similarly to the rest across the dataset but has the highest AUC accuracy for RGB data with occlusions.

We also note the effect of depth on the performance. Results across all filter

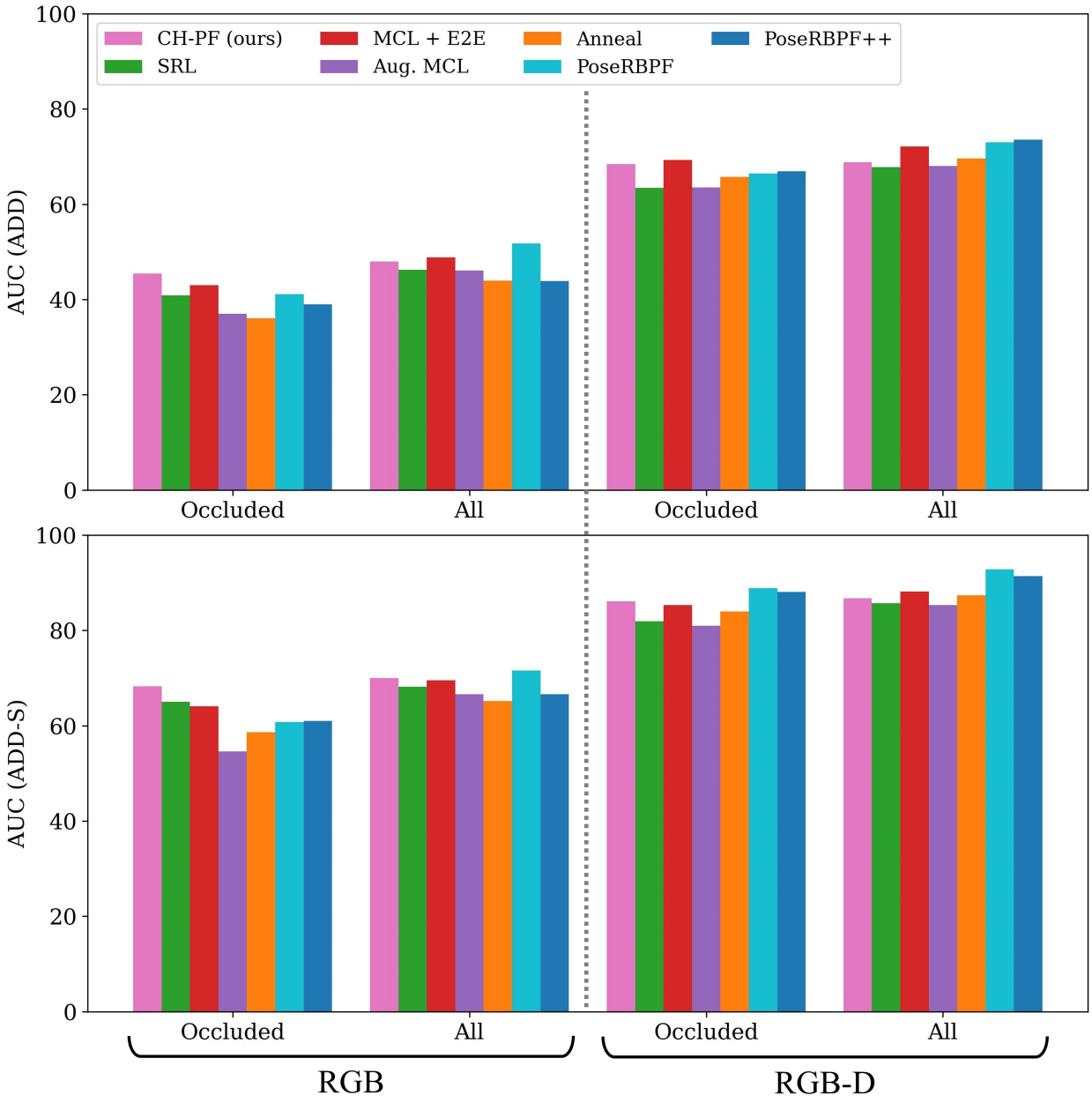


Figure 3.3: Area Under the Curve scores for all methods for object 6D pose tracking on the YCB Video Dataset. ADD scores and the symmetric version (ADD-S) are presented. Our presented method, CH-PF, has nominal performance across the scenarios but has notable improvement for RGB sequences with occlusion.

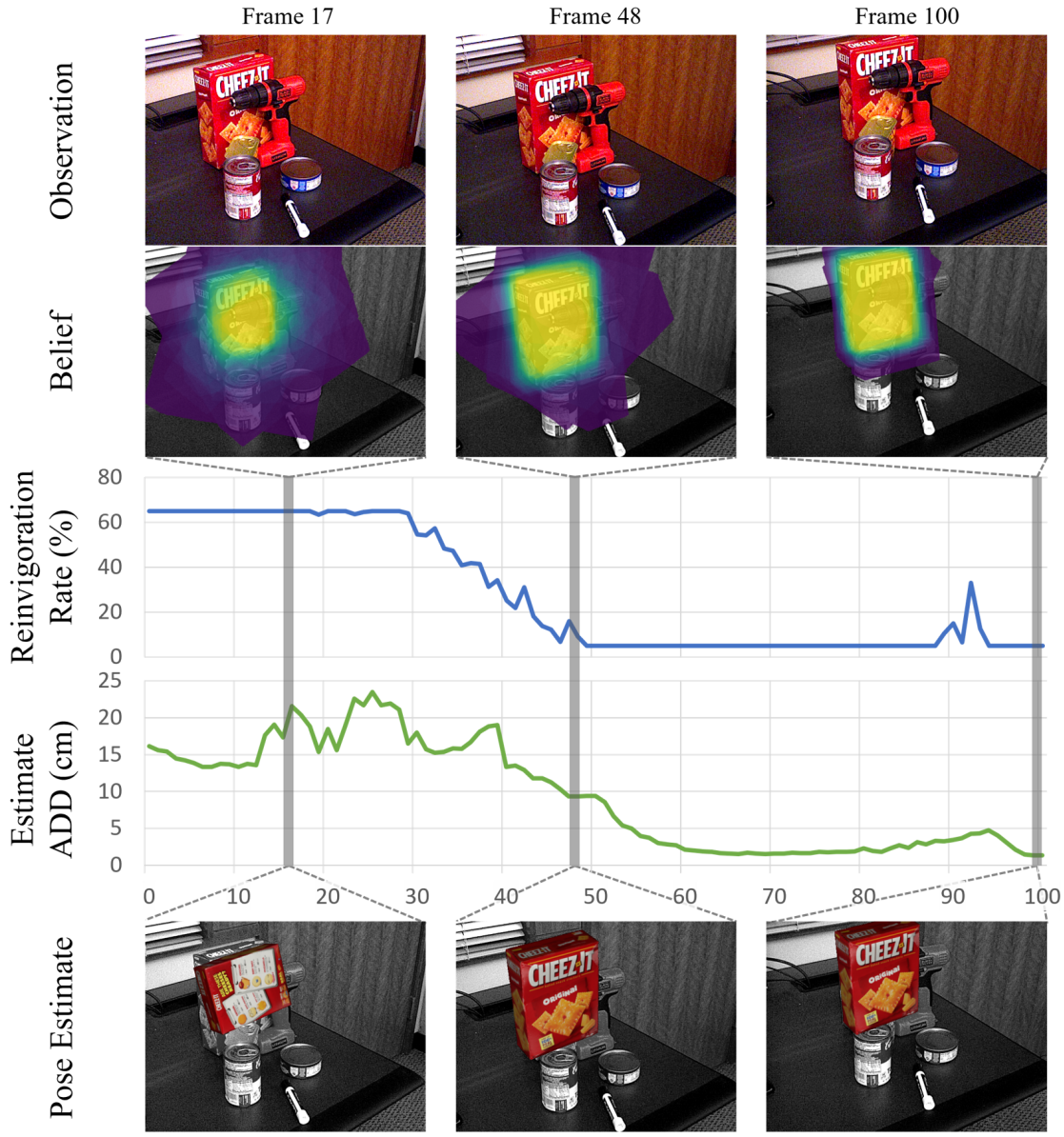


Figure 3.4: Selected qualitative results for the counter-hypothetical particle filter. The cracker box is significantly occluded by other objects in the scene. In early iterations (left), the cracker box belief is not converged, and the estimate has a high error. The reinvigoration rate, calculated from the belief, is high. In later iterations (middle), the belief converges to the ground truth state and the reinvigoration rate drops. The reinvigoration rate is low once the belief has converged (right). This figure is best viewed in color.

variants were improved when using RGBD data over RGB. By including depth, we hypothesize the likelihood function from PoseRBPF was more accurate and, therefore, a more useful indicator of when the filter was failing.

Selected qualitative results for the counter-hypothetical particle filter are shown in Figures 3.4 and 3.5. In these examples, the particle filter converged to an incorrect estimate at the beginning of the sequence. While the error was high, the counter-hypothetical particle filter could perform continued global localization with a high percentage of particles that performed a coarse search. Once the error dropped and a plausible region was found, the reinvigoration rate was reduced to focus its resources on exploring the space more closely.

The main limitation of our work is the additional computation time used at test time to evaluate each sample through an additional likelihood function. For simple heuristics, this would not add much time. In our case of another neural network, it doubles the inference time of extracting embeddings from the observation. With our computing setup and a particle set size of 50, the original PoseRBPF implementation can run at $30Hz.$ and our counter-hypothetical particle filter at $15Hz.$ Our performance was similar to that of using a single likelihood function for most of the dataset, but improvements in performance for heavily occluded scenes are promising.

3.7 Conclusion

This work aims to improve the accuracy of particle filters in tracking the 6D pose of rigid objects by adapting the rate of particle reinvigoration based on the estimated incompleteness of the current belief distribution. We propose independently estimating the potential error in each samples through a novel counter-hypothetical likelihood function. This modification allows us to reason over the cumulative doubt in our particle set, and use this estimate to apply particle reinvigoration as needed.

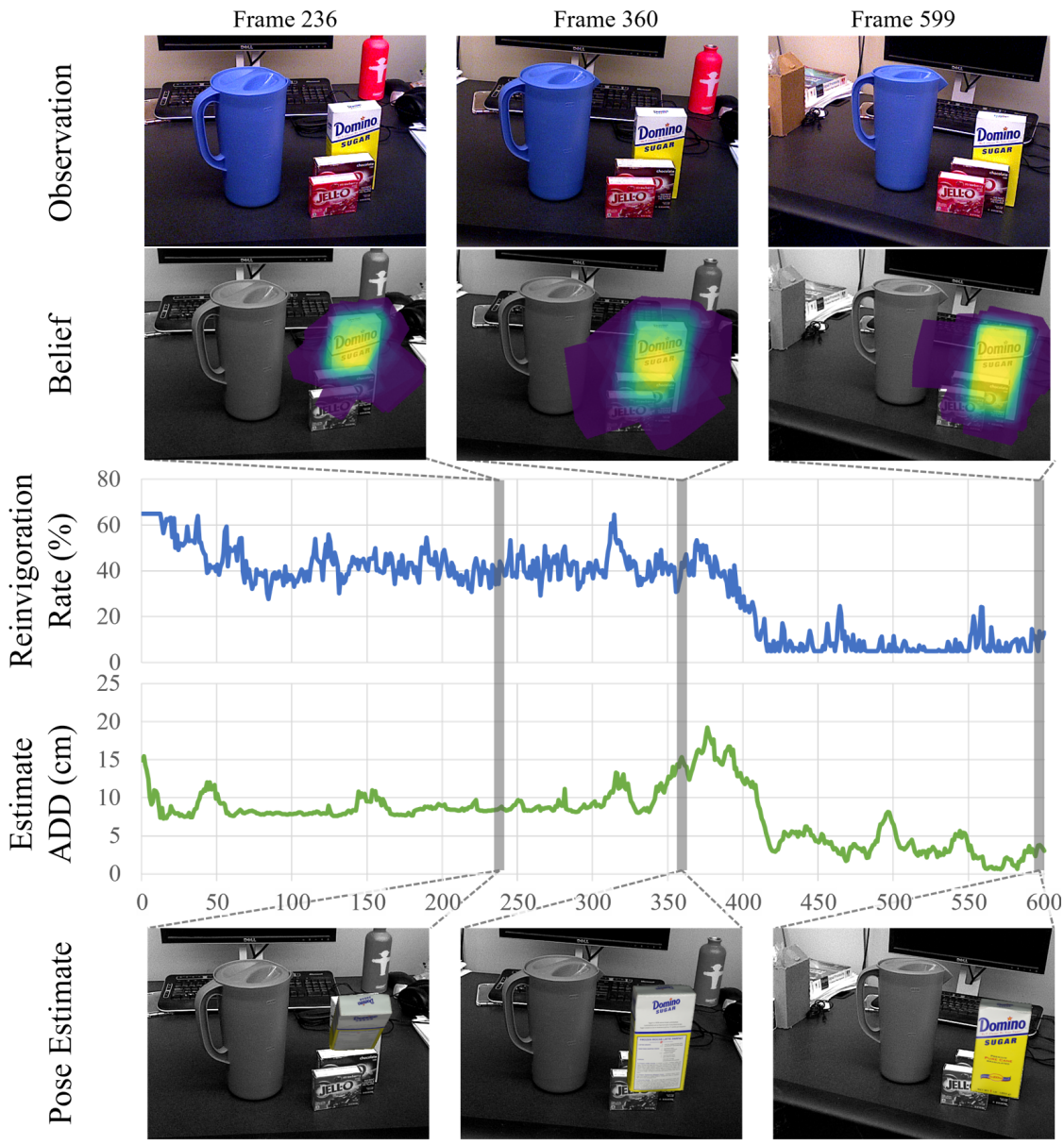


Figure 3.5: Selected qualitative results for the counter-hypothetical particle filter. The belief of the sugar box converges to a local maximum in early frames (left). CH-PF applies a higher reinvigoration rate to mitigate this. The error in the estimate briefly increases (middle), but the belief eventually converges to the correct estimate (right).

This chapter demonstrates the effectiveness of this modification as it matches overall performance when compared to standard methods of overcoming particle deprivation. Moreover, our particle filter proposed modification improves performance for scenes in heavy occlusion when only RGB data is present.

Chapter 4

The Progress Looking Upon a Moving BipEdal Robot (LUMBER) Dataset

Visual robot tracking allows us to compare tracking methods on a moving and highly articulated known object. In this specific application, methods can exploit the geometric information of a known model to overcome regions of the robot that may be visually obstructed. Many works have presented datasets for this task, but they are typically scenes with high observability or synthetically generated. We contribute a new dataset of 100 15-sec sequences of the humanoid robot, Digit [131]. Our dataset requires no external fiducial or motion capture markers and features heavy occlusions (only 10% of the dataset is unoccluded). It also contains a variety of obstacles, including fast, whole-body occlusions, static partial occlusions, and materials similar to Digit's parts. We also release this dataset to be a benchmark for the tracking community (http://lizolson.dev/progresslumberdataset).

4.1 Introduction

We present the Progress Looking Upon a Moving BipEdal Robot (LUMBER) Dataset. Its sequences feature a bipedal humanoid robot moving through an obstructed scene. We use a combination of manual and automatic labeling techniques to provide annotations of each link location. Our dataset contains 100 sequences of 15 seconds each ($>$ 44k frames). It includes static and dynamic obstructions, as well

as variations in the size of the obstruction.

4.2 Motivation

As robots move about homes and warehouses, they must avoid collisions with nearby agents, such as humans or colocated robots. Robust tracking of such agents is therefore crucial, despite the occlusions in real-world environments. To overcome this challenge, we must validate and improve visual tracking methods to ensure reliability. This requires tracking datasets containing the types of heavy occlusions representative of these work areas. Human pose estimation has been extensively studied, but robot pose tracking remains a research area of emerging datasets. It is also an application that presents the challenge of tracking a moving, high-dimensional object while exploiting the accessibility of a known model.

Though real and annotated datasets for robot pose tracking exist, any ambiguity in the scene is caused by self-occlusions [132, 133]. Recent work has focused on synthetic datasets, which allow for scalable annotation of tracking sequences and many scenes and obstacles to render [133, 134]. However, these datasets mainly present synthetic training data and very limited real testing data, if presented at all. To better test the sim-to-real gap and robustness to real-world environments, we present a benchmark dataset of a robot moving in realistic scenes with heavy occlusion. The Progress LUMBER Dataset was created to address this need through sequences of a bipedal humanoid robot moving through scenes of static and dynamic obstructions, as presented in this chapter.

(a) Example image from the CRAVES [133]

(b) Example from the DREAM [132]

Figure 4.1: Examples of RGB images from previous real-world collected robot pose datasets.

4.3 Related Works

4.3.1 Real-World Robot Tracking Datasets

A limited amount of real-world robot pose datasets have featured the annotation needed for deep learning pose estimation or tracking. The CRAVES [133] dataset captures a table-top manipulator moving in an unoccluded space, containing only a few thousand images. The DREAM [132] dataset contains a moving Panda manipulator, with images captured from several camera sensors. More comprehensive, this dataset contains over $50k$ images. However, it should be noted that these datasets only capture a few sequences, and there are no obstacles in the scene– meaning any obstruction is caused by self-occlusions. These characteristics make it difficult to trust the robustness of a tracking algorithm to poor initialization and partial observability. Examples from these datasets can be seen in Figure 4.1.

4.3.2 Synthetic Robot Tracking Datasets

From a deep learning perspective, tracking highly articulated objects such as robots is more data-hungry than tracking 6DoF rigid objects. With highly articulated objects, not only do various viewpoints of the object need to be represented in training data, but a sampling of the various configurations as well [73]. Synthetic

(a) Example image from CRAVES [134]

(b) Example from DREAM [132]

Figure 4.2: Examples of RGB images from previous synthetic robot pose datasets.

datasets are heavily used for training to address the need for such a large amount of annotated data.

Examples of the images in these synthetic datasets are displayed in Figure 4.2. The CRAVES [133] features synthetic images of its table-top manipulator moving about an unoccluded space. The DREAM [132] dataset also features synthetic images of several robots while also rendering obstacles to better handle occlusion. A more realistic synthetic image is produced by the Robot Tracking Benchmark [134] dataset, which takes advantage of recent improvements in rendering.

4.4 Dataset Collection

For data collection, we captured sequences of Digit moving in front of an Azure Kinect Sensor. RGB and depth images of the scene were recorded. To be able to later annotate the dataset, we also recorded the joint values estimated by Digit's encoders, as well as a base link pose estimation provided by an onboard Extended Kalman Filter(EKF). The coordinate frame Digit's EKF has an origin at Digit's starting position, which is not the same origin as the world frame captured by the camera. The RGB image was then synced with the depth image, joint values, and EKF estimate temporarily closest to it.

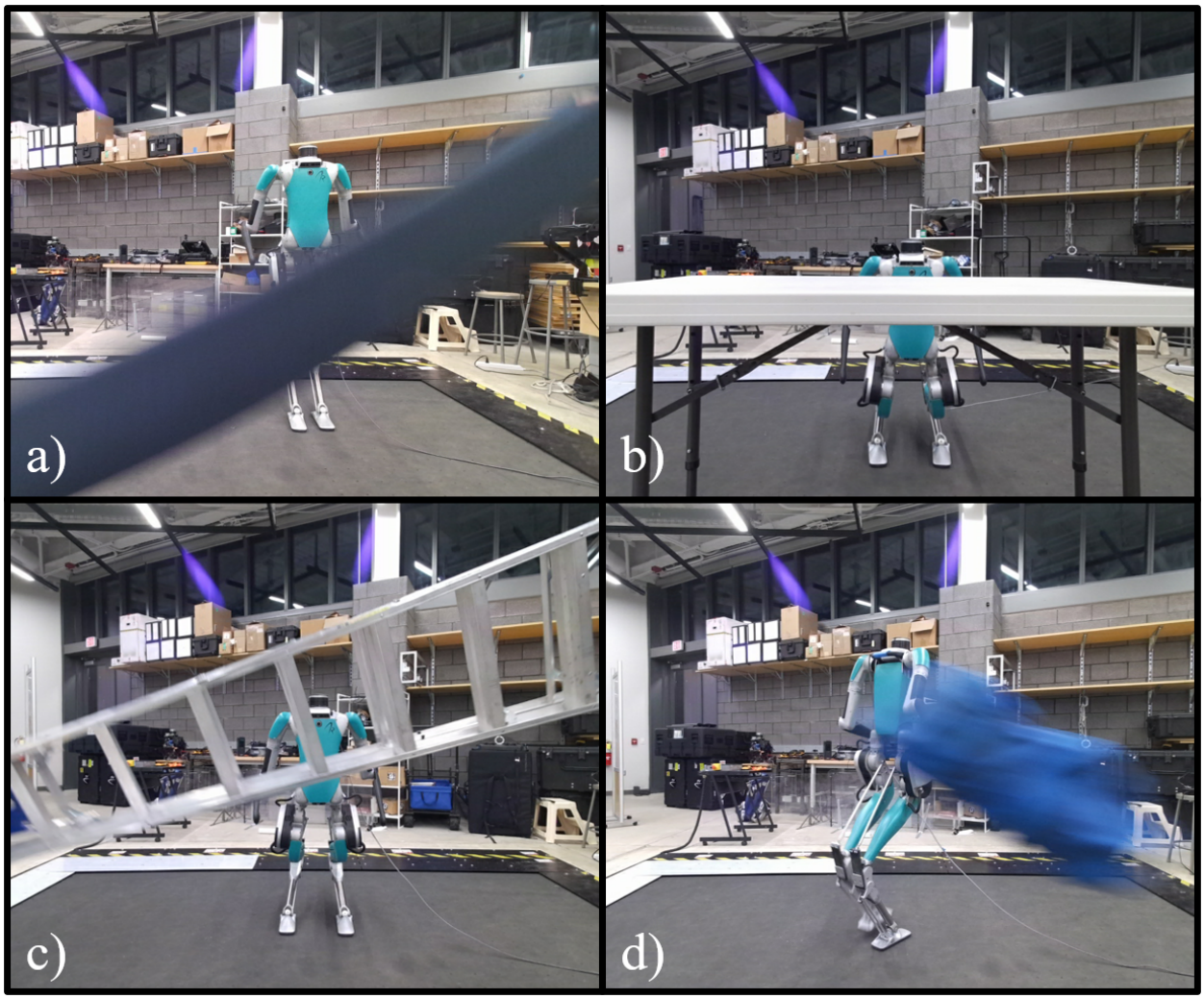


Figure 4.3: We present a humanoid robot tracking dataset featuring occlusions in 90% of the sequences. These occlusions occur from both dynamic and static obstacles, such as a) a shaken cloth, b) a static table, c) a moving metallic ladder, and d) thrown bags and buckets.

The sequences in the dataset featured a variety of movements and motions from Digit. Digit moved forward and backward, laterally, turning, and popping up from a lying down position. To test the robustness of tracking algorithms to fast movements, its lateral and sitting-up motions had the highest velocity for the base link. The turning and laying down positions also gave unique viewpoints of the robot not often featured in videos. Each setup also included multiple sequences of the robot not walking, but rather twisting its torso and moving its arms.

As is documented in Figure 4.3, the setup of the sequences for the dataset

Setup Name	Type	Size	Occlusion	Frames
bags	dynamic	medium	full	4451
cloth	dynamic	full-body	full	4452
ladder-moving	dynamic	medium	partial	4451
ladder-still	static	medium	partial	4448
pole	dynamic	small	full	4443
shelf	static	medium	partial	4438
table	static	medium	full	4451
unoccluded	-	-	-	4439
whiteboard-narrow	static	medium	full	4444
whiteboard-wide	static	full-body	full	4450

Table 4.1: Catalogue of the ten sequence setups in the dataset. There are ten sequences collected for each setup.

included many different obstructions. Some obstacles, such as a set of thrown bags, moved cloth, a ladder, and a pole, were dynamic objects that quickly changed the observability of Digit. Other sequences contained static objects, such as a ladder, table, whiteboard, and shelf. We also note that some of these obstructions were continuously blocking the background (like the whiteboard and table), while others featured partial observability (mainly the shelf and ladder). In Table 4.1, we give more information about each scene setup. Each setup was used for ten sequences. These were comprised of two sequences of lateral walking, two sequences of forward-backward walking, two sequences of walking while turning in place, two sequences of hip and torso movements without walking, and two sequences of Digit lying on the ground and then popping to a standing position.

4.5 Dataset Labeling

4.5.1 Coarse Manual Registration of Sequence

For each of the 100 sequences, we must have an initial pose annotation, the process of which can be overviewed in 4.4. Using the rendering software Blender,

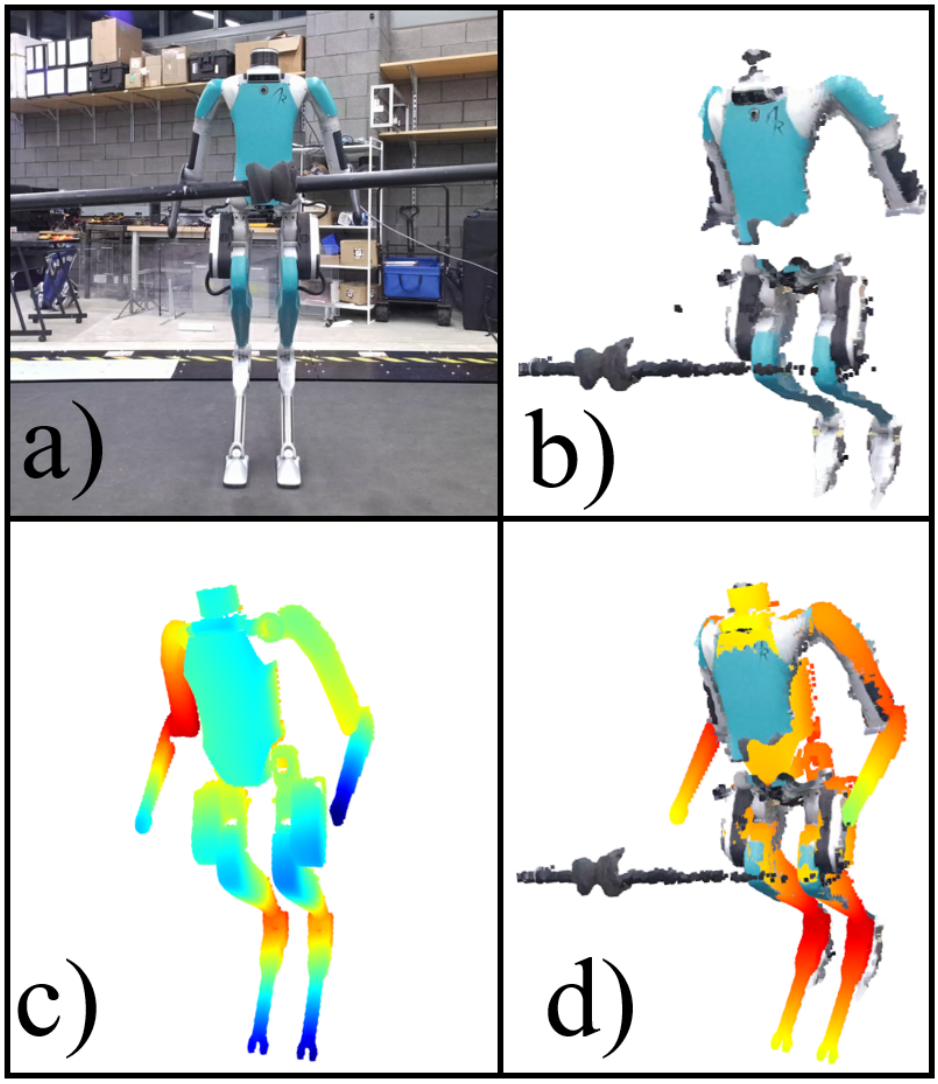
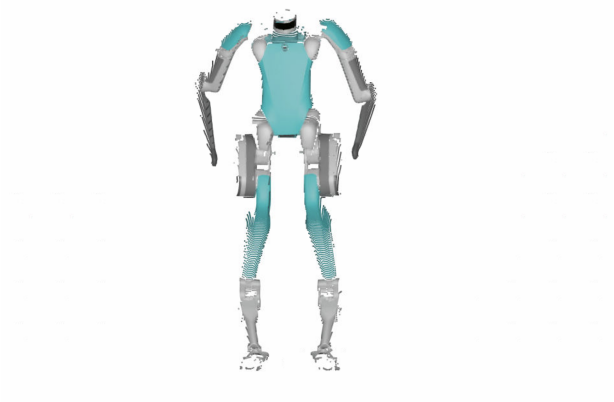


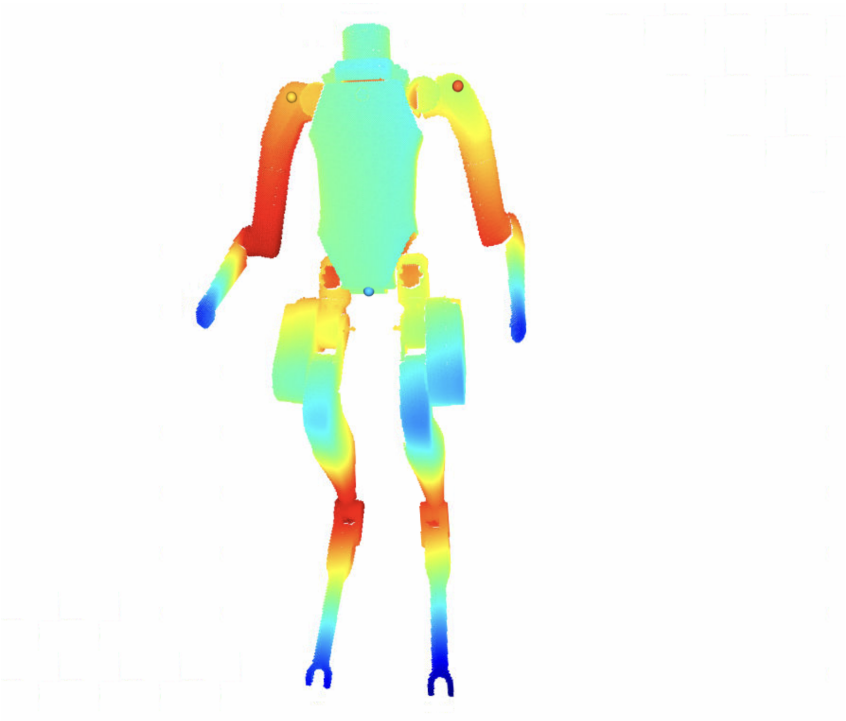
Figure 4.4: An example of the labeling process for our contributed dataset. a) We record Digit moving with an RGB-D sensor from the dataset that is then b) converted to a point cloud of the scene. c) The recorded joint configuration of Digit is rendered. d) Using iterative closest point (ICP) [5], the transformation of the base link is found to compute the location of all joints with respect to the RGB-D sensor's frame.



(a) Plane of point cloud when rendered from a front view

(b) Plane of point cloud when rendered from a side view

Figure 4.5: For a captured point cloud to match a rendered point cloud for manual registration, a rough orientation of the robot needs to be input. Examples of changes in the plane of the point cloud when rendered from different viewpoints



(a) Plane of point cloud when rendered from a side view

(b) Plane of point cloud when rendered from a side view

Figure 4.6: For a captured point cloud to match a rendered point cloud for manual registration, a rough orientation of the robot needs to be input.

the robot is rendered in the configuration of the recorded joint values. The depth image produced is converted into a point cloud to be registered within the captured point cloud of the scene. Since the captured point cloud only includes points in the scene visible to the camera, the synthetic point cloud must include the same respective regions of the robot. Otherwise, the registration algorithm may misalign the point clouds. As illustrated in Figure 4.5, a rough orientation estimation is used.

Once a point cloud similar to the scene is registered, it must be roughly aligned to the scene. For this, manual point cloud registration is performed. As shown in Figure 4.6, each point cloud is manually labeled with at least three correspondence points. Then, a point-to-point implementation of iterative closest point (ICP) [135] from the Open3d library [5] registers the rendered point cloud within the captured point cloud.

4.5.2 Frame-Level Fine Annotation

We then looked at three different pipelines to propagate the coarse manual annotation for the sequence to fine, frame-level annotations. We could not compare them quantitatively because we did not have ground truth annotations. Instead, we show examples of the RGB frame paired with the rendering of an annotated pose overlayed on the RGB frame. In this Section 4.5.2.3, we describe the pipeline ultimately used, which combined visual and proprioceptive information.

4.5.2.1 ICP-Only Annotation

At the first attempt, the coarse manual registration gave a rough estimate of the pose to be corrected by ICP [136]. The current configuration was rendered to a depth frame and converted to a point cloud, similar to the manual registration. However, the registration was fine-tuned by the ICP algorithm. Unfortunately, this

Figure 4.7: An example of a captured image (left) and its corresponding annotation (right) when the clouds are automatically registered at each frame. The annotation is too sensitive to changes in the visibility of the robot, resulting in annotation across the sequence having too much jitter.

method was susceptible to a lot of noise and jitter between consecutive frames, as shown in Fig 4.7. When the observability of Digit significantly changed between frames, such as when the robot was moving laterally behind an obstacle, or a thrown bag occluded the torso very quickly, the annotation became significantly wrong.

4.5.2.2 EKF-Only Annotation

Our next strategy was to use the filtered pose from the Extended Kalman Filter (EKF) running onboard the robot. Visualizing its pose, it was very smooth and realistic. The coordinate frame of the EKF pose could be transformed to the coordinate frame of the captured image (which had the camera at its origin) by finding the transformation between the EKF pose and the manually labeled pose. However, though the sequence was only 15 seconds, that was enough time for drift to occur in the filter's pose, as it did not have any corrections from visual information. Such drift is depicted in Fig 4.8.

Figure 4.8: An example of a captured image (left) and its corresponding annotation (right) when the pose given by the EKF data is used throughout the sequence. Since the filter uses no visual information, drift occurs over time.

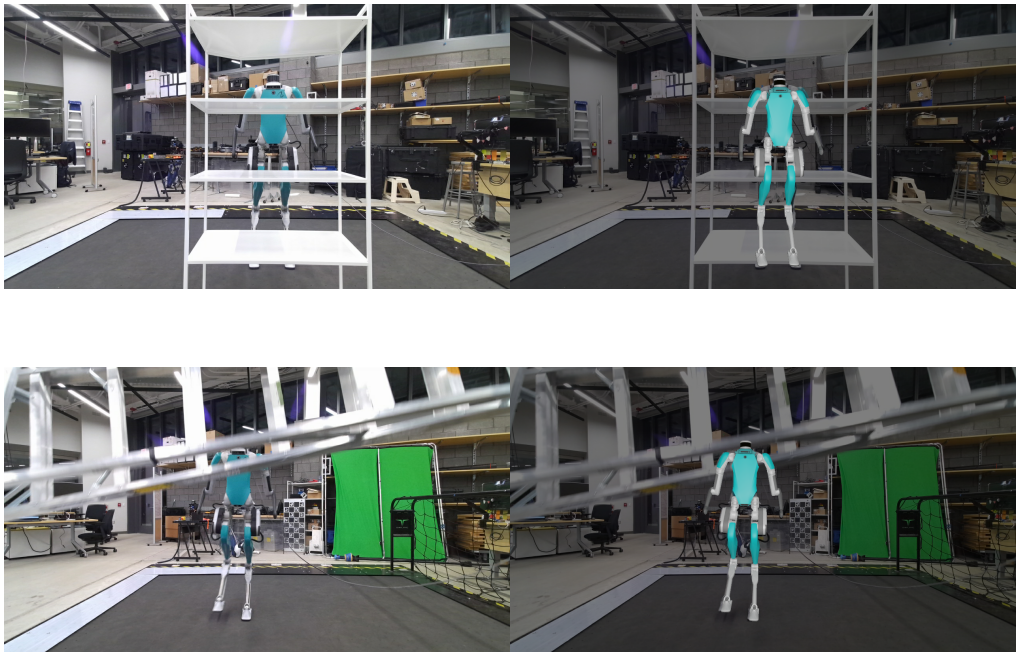


Figure 4.9: Final annotations from the provided dataset

4.5.2.3 ICP and EKF Annotation

The best results were achieved through a combination of the previously mentioned methods. Manual registration gave the transformation between Digit's internal frame and the world frame. At each frame, we had an initial estimate of Digit's pose in its internal frame from EKF. The manual transformation with the frame's pose from EKF provided an initial estimate of Digit's pose in the world frame. We needed to use visual information to reduce drift, but ICP at each frame was too noisy. We created a set of potential keyframes that refined the initial pose from EKF. The keyframes were every 25 frames, making them less than one second apart. We then used the depth image of the scene to refine the initial pose estimate through ICP. These keyframes were often not useable and had to be manually removed. By projecting the estimated pose on the RGB frame, we could remove keyframes that were significantly wrong.

With a reduced set of keyframes, we needed to combine the poses of the keyframes with the EKF data for smooth and consistent pose estimations across the sequence. To do this, we formulated the final pose of the robot as the original EKF pose, transformed based on the EKF-to-image-pose we had calculated, and then a final transformation to account for the drift over the sequence. The keyframes told us what the drift of the previous timesteps would sum to, but we needed to interpolate the drift in between. We used a spherical linear interpolation (slerp) [137] to estimate the drift between keyframes. As shown in Figure 4.9, these annotations were smooth and filtered between timesteps and free from drift across the sequence.

4.5.3 Label Generation

At this point in the annotation pipeline, we had the 6DoF pose of the base link, the torso, as viewed by the camera. We rendered the base at this pose and calculated the necessary positions of its neighboring links via forward kinematics.

Blender also allowed us to label the segmentations and bounding boxes for each link. We paired these labeled poses with the original RGB and depth images and recorded joint configurations to release them to a larger audience.

4.6 Conclusion

In this work, we sought to create a dataset representative of a task we may soon see in the real world: highly articulated moving objects in scenes of clutter and occlusion. We collected and labeled a 100-sequence visual tracking dataset for a humanoid biped. We separated its contents from similar datasets by featuring setups in which the robot had partial observability—whether through external obstacles or self-occlusion. We used a combination of internal filtered data from the robot's proprioceptive sensors and depth information captured from the camera to produce the best labels. We provide 6DoF pose for each link, segmentations, joint configurations, and bounding boxes for over 44k frames.

Chapter 5

WAGER-DNBP: Weighted And Graphical Evidential Reasoning for Differentiable Nonparametric Belief Propagation

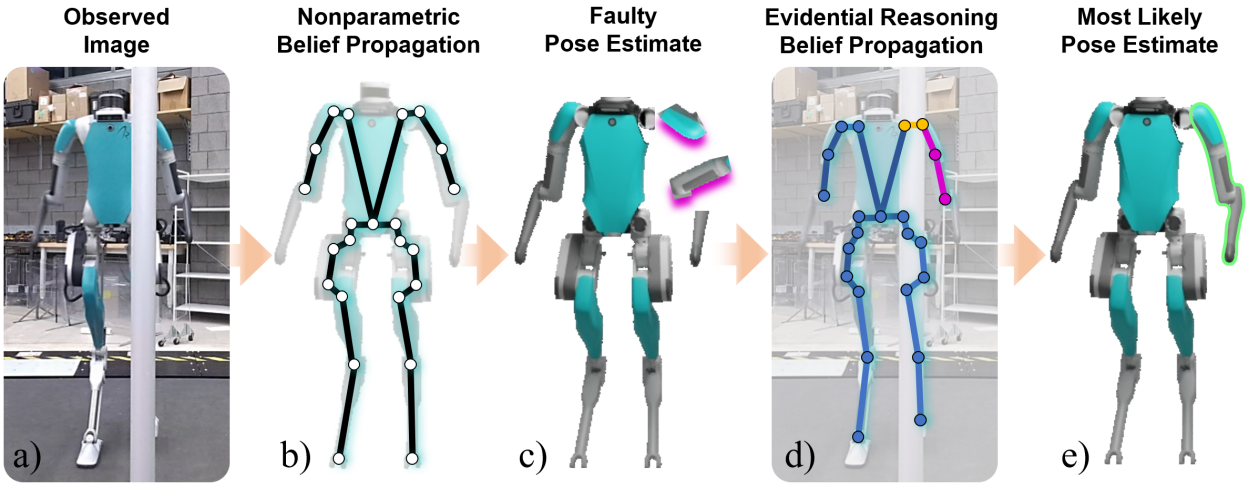


Figure 5.1: (a,b) The observed humanoid robot (RGB version shown) is represented as a graphical model for our nonparametric belief propagation representation. (c) The current estimate is very off for the left arm. d) We use evidential reasoning to estimate that the left shoulder's observation is ambiguous, and the nodes of the left arm are in failure mode. e) This conclusion informs our resampling and redistribution of samples in the left arm for faster recovery.

In this chapter, we introduce **W**eighted **A**nd **G**raphical **E**vidential **R**easoning for **D**ifferentiable **N**onparametric **B**elief **P**ropagation, WAGER-DNBP as an extension of nonparametric belief propagation beyond Bayesian reasoning to consider ambiguity and disbelief — which we hypothesize is beneficial for the occlusions present in real-world tracking. Tracking highly articulated robots presents a particularly motivating challenge due to the intricate state space and potential partial observability.

Integrating data-driven modeling in a reliable probabilistic framework is useful, and differentiable nonparametric belief propagation is one such method. However, random sampling of the technique occasionally produces incorrect estimates when the filter fails, and there is no dependable signal to redistribute the samples. To tackle this issue, we introduce Deep Evidential Reasoning to nonparametric belief propagation to better handle noisy and error-prone tracking scenarios. Our approach is validated on our previously presented humanoid robot tracking dataset featuring major occlusions.

5.1 Introduction

As robots progress toward operating autonomously in unstructured and collaborative environments, they are increasingly expected to be aware of the location and movements of nearby agents. Sometimes, these robots must rely exclusively on visual perception to track human collaborators or co-located robots. Meeting this expectation requires accurate tracking algorithms that remain efficient when tracking highly articulated objects, especially in situations with heavy clutter or quick movements.

Differentiable Nonparametric Belief Propagation (DNBP) has demonstrated promising success due to its effective combination of powerful data-driven models with robust and diagnosable probabilistic inference [6]. Despite these benefits, the time constraints placed on robotic applications necessitate limited computation and small sample sets within DNBP. This restriction can prevent the algorithm from sampling near the true state and instead waste computation exploring implausible regions of the state space. This failure mode could be alleviated by accurately identifying the well-observed regions and weighting their information higher, as well as identifying the poor-performing nodes and redistributing their samples to recover the true belief. Because nonparametric belief propagation is founded in Bayesian reasoning, analysis of whether the node is failing can only be indirectly inferred by noticing the low likelihood scores. But, this raises an important question: is such a low likelihood

due to to the inaccuracy of the hypothesis or due the model's inability to reason about *any* hypothesis given the vantage point?

Evidential reasoning is an emerging field within deep learning that aims to quantify uncertainty and model ambiguity of neural network-based algorithms. This paradigm distinguishes between uncertainty due to noise or ambiguity and uncertainty due to glaring error, making it well-suited for augmenting factor graph models such as DNBP. Many works have integrated neural networks into probabilistic filters, but mainly by using the scores of the network as likelihood functions for the confidence in the sample's accuracy.

Our key insight is that a model's estimated uncertainty in its own performance, provided by evidential reasoning, can be embedded into a Bayesian nonparametric belief propagation. These confidences can then be used to create a hierarchy of the visual information being propagated through the network to enable faster recovery.

5.2 Related Works

5.2.1 Deep Uncertainty Quantification

In an attempt to mitigate the inherent opaqueness of deep neural networks, uncertainty quantification provides a path towards diagnosability and transparency [138, 139]. It seeks to measure the risk associated with a given hypothesis—an important consideration as these estimates can inform downstream robotic tasks that are becoming increasingly critical [140] [141]. Uncertainty is generally categorized as either due to the model's need for further exposure to unfamiliar data (epistemic) or from the noise and unpredictability intrinsic to the data (aleatoric) [94, 142]. Given neural networks' sensitivity to inferring on samples outside the distribution of their training data, epistemic uncertainty has garnered extensive research in the field for out-of-distribution detection [92, 109, 143]. Aleatoric uncertainty, applicable for reasoning about partially observable scenes, is often quantified by computationally expensive ap-

proaches, such as measuring the consensus of an ensemble of networks [101, 102], or by analyzing the consistency of output from a single stochastic neural network[144].

In deep evidential reasoning, the network explicitly and internally models its ignorance at the time of inference. It is grounded in Evidential Theory [24], a generalization of Bayesian reasoning that not only quantifies likelihood but additionally measures disqualifying information, as well as the ignorance and ambiguity of reaching any conclusion. Examples include augmenting a classification set with an 'unknown' category [108] or learning to estimate the parameters of an underlying distribution to measure uncertainty in regression applications [111]. The theory's independent quantification of counter-hypothetical information, or doubt, has also been integrated into a particle filter to inform resampling [25]. This chapter paper presents an approach to learning to measure the supporting, aleatoric-based, ambiguous, and counter-hypothetical information associated with a possibility and extend Bayesian graphical models to propagate the information. While previous deep evidential reasoning implementations inform a human supervisor or downstream planning tasks, we present an integrated framework to improve the quality of the perception itself.

5.2.2 Nonparametric Belief Propagation

Nonparametric belief propagation (NBP) is an established method for tracking highly articulated objects in continuous state spaces [79, 10, 11, 12]. In contrast to traditional sum-product belief propagation [9] that requires exact integral computations, NBP algorithms approximate continuous posterior distributions using graph-based message passing with discrete sample sets. For an articulated object of interest, these approaches encode the known articulation constraints in a factor graph representation, then use local message passing operations to infer the posterior distribution over each part's pose given access to some observed sensor data (e.g., images from a

camera) [145]. The decoupled structure of NBP allows for faster and simpler training than fully regressing high dimensional pose [73]. It can focus on regions of the object that are observable for better local pose estimation.

Unfortunately, like all sampling-based inference methods, the algorithm can struggle to fully represent such a large state space when constrained to a low number of particles. Many works, particularly those from the Monte Carlo localization community, have looked to reinitialize samples as needed in particle filters, known as adaptive particle reinvigoration [85, 87, 84, 128, 112]. However, research has yet to extend adaptive particle reinvigoration to NBP by estimating the appropriate frequency to sample from the *multiple* proposal distributions available in NBP. Instead, these methods maintain particle diversity by drawing from the different distributions at fixed ratios determined by hand-tuning [146, 13]. Though the resampling step is not truly differentiable for filters trained as neural networks [16, 17, 6], our work explores learning auxiliary signals via Evidential Reasoning to determine when particle redistribution is necessary.

5.3 Methodology

Given a sequence of t depth images, $z_{1:t}$, we seek to localize the 6D pose, x_{st}, of an link s at time t. The marginal belief distribution of X_s at time t, $bel^t(X_s)$, can be approximated by

$$bel^t(X_s) \propto \phi_s(X_s, Z_s) \prod_{r \in \rho(s)} \hat{m}^t_{rs}(X_s) \tag{5.1}$$

where $\phi_s(X_s, Z_s)$ is the unary potential of the latent state X_s and its corresponding observable state, Z_s. $\hat{m}^t_{rs}(X_s)$ represents the message passed from r to s, where r is a neighboring node of s as indicated with $r \in \rho(s)$. The messaged passed from r

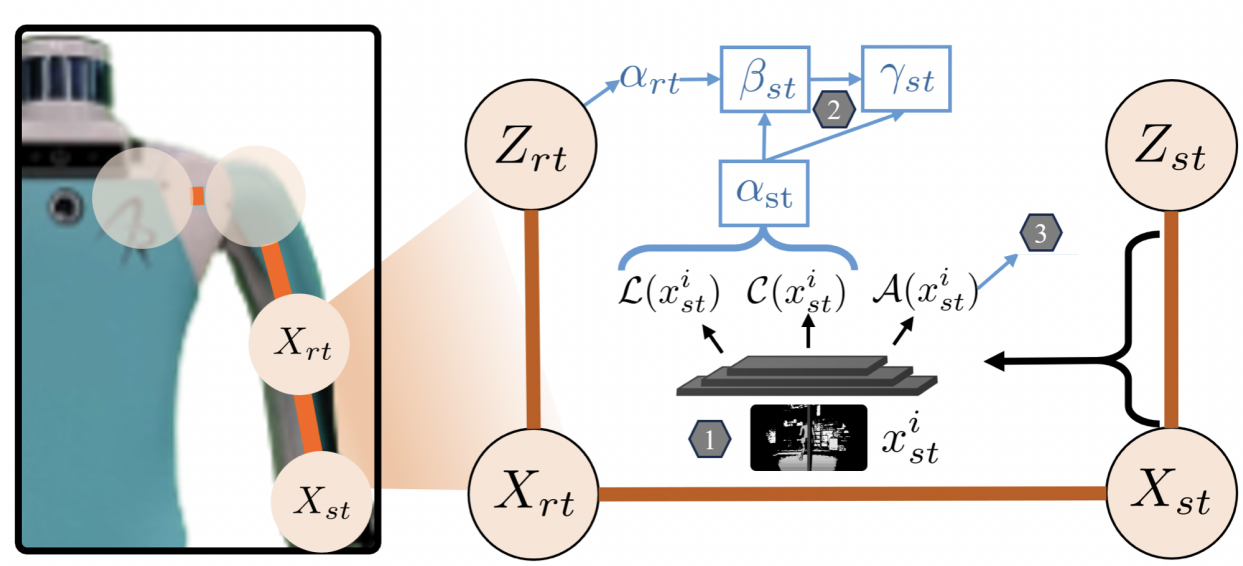


Figure 5.2: Explanation of our proposed observation model that leverages deep evidential reasoning: 1) The hypothesis and depth image are passed into the network, which assigns three weightings to the sample: likelihood, counter-hypothetical likelihood, and ambiguity. 2) The aggregate of the unnormalized likelihood scores and counter-hypothetical likelihood scores across the particle set determine α_{st}, a measure of the node's overall performance. The α values of the given node and those of its neighbor(s) determine the ratio of samples to be drawn from the prior distribution (α_{st}), neighboring distributions (β_{st}), and random distribution (γ_{st}). 3) The ambiguity score of each sample will be used in a weighted sum calculation for the messages passed from X_{st} to its neighbors.

to s at is defined as:

$$\hat{m}_{rs}^{t}(X_s) = \sum_{X_r \in \mathbb{X}_r} \phi_t(X_r, Z_r)\psi_{r,s}(X_r, X_s) \prod_{u \in \rho(r) \setminus s} \hat{m}_{ur}^{t}(X) \tag{5.2}$$

These equations demonstrate the chain of messages passing into a given node through belief propagation. Specifically with nonparametric belief propagation, the belief distribution of $bel^t(X_s)$ is represented by a set of particles $\mathbb{X}_s$:

$$\mathbb{X}_{st} = \{(x_s t^1, \pi_s t^1), (x_s t^2, \pi_s t^2), \ldots, (x_s t^N, \pi_s t^N)\}, \tag{5.3}$$

where x_s^i is the ith sample of the particle set, and π_s^i is its corresponding normalized importance weighting, given from Equation 5.1, and N is the number of particles at the given node. In traditional nonparametric belief propagation, the next set of particles would be sampled off of the current set, and the probability of a given particle being selected would be based on its importance weighting. However, this causes mode collapse in the underlying belief distribution, pushing the filter into failure mode. In practice, $\mathbb{X}_s$ for the next iteration is often a combination of samples from the current set, $\mathbb{X}_s^{prop}$, as well as randomized particles sampled off a set of sampled off of other candidate proposal distributions, $\mathbb{X}_s^{aug}$. With $\mathbb{X}_s = \mathbb{X}_s^{prop} \bigcup \mathbb{X}_s^{aug}$, the ratio from which to sample off of each distribution needs to be addressed.

Similar to the counter-hypothetical particle filter [112], our observation model outputs more weightings than just the typical likelihood score. Previously, a counter-hypothetical likelihood and likelihood function were applied within a particle filter to determine the number of samples to be drawn from a random distribution for adaptive particle reinvigoration. However, with nonparametric belief propagation, information is passed between multiple nodes, and more analysis can be considered through evidential reasoning. From the counter-hypothetical particle filter, we similarly

estimate α_s, the ratio of samples from $\mathbb{X}^{prop}$ for the particle set at the s node:

$$\alpha_{st} = \frac{\sum_{i=1}^{N} \mathcal{L}(x_{st}^i)}{\sum_{i=1}^{N} \mathcal{C}(x_{st}^i) + \sum_{i=1}^{N} \mathcal{L}(x_{st}^i)} \tag{5.4}$$

where $\mathcal{L}(x_t^i)$ is the unnormalized likelihood weighting for the given particle, and $\mathcal{C}(x_t^i)$ is the unnormalized weighting from the counter-hypothetical likelihood. Note that the ratio of particles sampled from distributions other than the previous particle set, comprising $\mathbb{X}^{aug}$, would be $1 - \alpha$. This formulation is incomplete for adaptive particle reinvigoration within belief propagation, as there are multiple candidate distributions from which the samples can be reset. They may be reinitialized from a random uniform distribution, similar to initialization, which we'll denote $\mathbb{X}^{rand}$. Otherwise, they may be sampled off of a distribution created from samples of the neighboring nodes [146, 147], denoted here as $\mathbb{X}^{pair}$. To extend the notation of particle reinvigoration to this case, we find $\mathbb{X}^{aug} = \mathbb{X}^{rand} \bigcup \mathbb{X}^{pair}$, leaving us to determine the ratio between $\mathbb{X}^{rand}$ and $\mathbb{X}^{pair}$.

We then introduce β, the ratio of augmented particles sampled from neighboring samples, $\mathbb{X}_s^{pair}$. Intuitively, this ratio should be in accordance with our confidence that the neighboring nodes contain plausible samples. Therefore, it is the average of the α scores of each of the M neighboring nodes:

$$\beta_{st} = (1 - \alpha_{st}) \cdot \frac{1}{M} \sum_{r \in \rho(s)} \alpha_{rt}. \tag{5.5}$$

For the particles reinvigorated in $\mathbb{X}^{pair}$, the frequency from which each neighboring node's belief is sampled from is proportional to its α score relative to the other neighbors.

The ratio of particles to be sampled from a uniform random distribution is defined as γ. This is then calculated from the other ratios and the fact that they

must sum to 1:

$$\gamma_{st} = 1 - \alpha_{st} - \beta_{st}. \tag{5.6}$$

The size of $\mathbb{X}_s^{prop}$, $\mathbb{X}_s^{pair}$, and $\mathbb{X}_s^{rand}$ are then $\alpha_s N$, $\beta_s N$, $\gamma_s N$ respectively. Note that $\alpha_s N$, $\beta_s N$, and $\gamma_s N$ will need to be rounded to integers that have a sum of N.

These proposed extensions to nonparametric belief propagation incorporate the notion of disbelief presented by Evidential Theory but could also benefit through their modeling of ambiguity. Aleatoric uncertainty quantification measures the noise and unpredictability of the observation, and we propose its implementation within a graphical model as a mechanism to evaluate the usefulness and trustworthiness of each node's unary potential. The third scalar weighting each observation model produces for each sample measures the ambiguity, $\mathcal{A}(x_{st}^i)$. For this quantity, the node learns to make a 'wager' that the unary potential is reliable for the given hypothesis and observation. We use $1 - \mathcal{A}(x_{st}^i)$ as the scaling factor in a weighted sum of the message to be passed to the node's neighbors. The less ambiguity a sample has associated with its unary potential, the greater its influence on the message's sum. For this functionality, we modify Equation 5.2:

$$\hat{m}_{rs}^t(X_s) = \sum_{X_r \in \mathbb{X}_r} (1 - \mathcal{A}(X_r))\phi_t(X_r, Z_r)\psi_{r,s}(X_r, X_s) \prod_{u \in \rho(r)\setminus s} \hat{m}_{ur}^t(X) \tag{5.7}$$

With this formalization, a node will mainly sample from its prior distribution when there is little evidence that its current particles are wrong. If a node has particles that are observed to be glaringly wrong and have neighboring nodes that appear to be performing well, the node's particles will be sampled off of particles at the neighboring nodes. Otherwise, if the neighboring nodes also appear to have poor performance, the node's particles will be initialized from random. In this way, incorporating evidential reasoning into the formulation of Bayesian nonparametric belief propagation can result in a faster recovery of the true belief.

5.4 Experiments

5.4.1 Implementation

Our implementation builds off of an open-source differentiable nonparametric belief propagation network(DNBP) [6]. In both the original DNBP and our variant, the network knows the connectivity of the joints but not the geometry. Instead of its observation model giving a single likelihood score to each particle, we alter it to score each particle with three scores(likelihood score, counter-hypothetical score, and ambiguity score). For the unary likelihood score and pairwise potentials, we utilize the original loss function, which maximizes the posterior distribution at the point of the ground truth state.

$$L_s = -log(\bar{bel}_s^t(x_s^{GT})) \tag{5.8}$$

where x_s^{GT} is the ground truth pose of the joint and $\bar{bel}_s^t(x_s^{GT})$ is the density at that point from a probabilistic density function formed of the weighted samples.

For training the counter-hypothetical likelihood function, we follow previous evidential reasoning works [111, 112] that aimed to train a network to identify errors in incorrect estimates. These works only penalize estimates significantly distanced from the true label. This loss function trains the second weighting value to identify glaringly wrong samples by maximizing the counter-hypothetical weighting of significantly wrong samples (those with a Euclidean distance greater than 0.1m).

$$L'_s = \sum_{x_{st}^i \in \mathbb{X}_{st}} \begin{cases} -\log(\mathcal{C}(x_{st}^i)) & \texttt{if } \|x_{st}^i - x_{st}^{GT}\| > 0.1m \\ 0 & \texttt{otherwise} \end{cases} \tag{5.9}$$

Similar to the confidence prediction from Segment Anything [148], we compute the compatibility between the likelihood and counter-hypothetical likelihood scores during training and use this as a supervisory signal for training the ambiguity score,

the final weighting provided by the observation model. We assume that when less aleatoric uncertainty is present, the likelihood and counter-hypothetical likelihood scores should be reliable and consistent. The likelihood and counter-hypothetical likelihood functions are ultimately passed through a sigmoid layer, so we calculate how close their sum is to 1. The ambiguity weighting learns to regress the error between their sum and 1.

The networks were trained and tested on our contributed Progress LUMBER Dataset. As our dataset has ten scene setups, we used eight for training and two for testing for a standard 80/20 train/test split. We used the two scene setups with the whiteboard, whiteboard-narrow (a side-view of a rolling whiteboard is occluding the robot) and whiteboard-wide (the full length of the whiteboard is occluding the robot), as testing data.

5.5 Results

We continue the error metric initially used for DNBP's analysis on hand-tracking [6]. This metric measures the error distance between the estimated and true joint locations. These are recorded across all joints in all frames of the test set. Similar to our work with the rigid object error metrics ADD and ADD-S, the aggregate of these results is analyzed by a Receiver Operating Characteristic (ROC) Curve.

We highlight a sequence of results from the dataset to show the performance of WAGER-DNBP over DNBP. During the evaluation of a lateral walking sequence, two time snapshots are examined at Fram 10 and Frame 74 (Figure 5.3). Our WAGER-DNBP method demonstrated consistent performance in maintaining accurate estimates throughout the sequence. On the other hand, the baseline model, which lacks the integration of evidential reasoning, displayed limitations. We see both methods can recover the pose of the robot shortly after initialization. However, once it passes behind a side-view of a rolling whiteboard, DNBP significantly loses

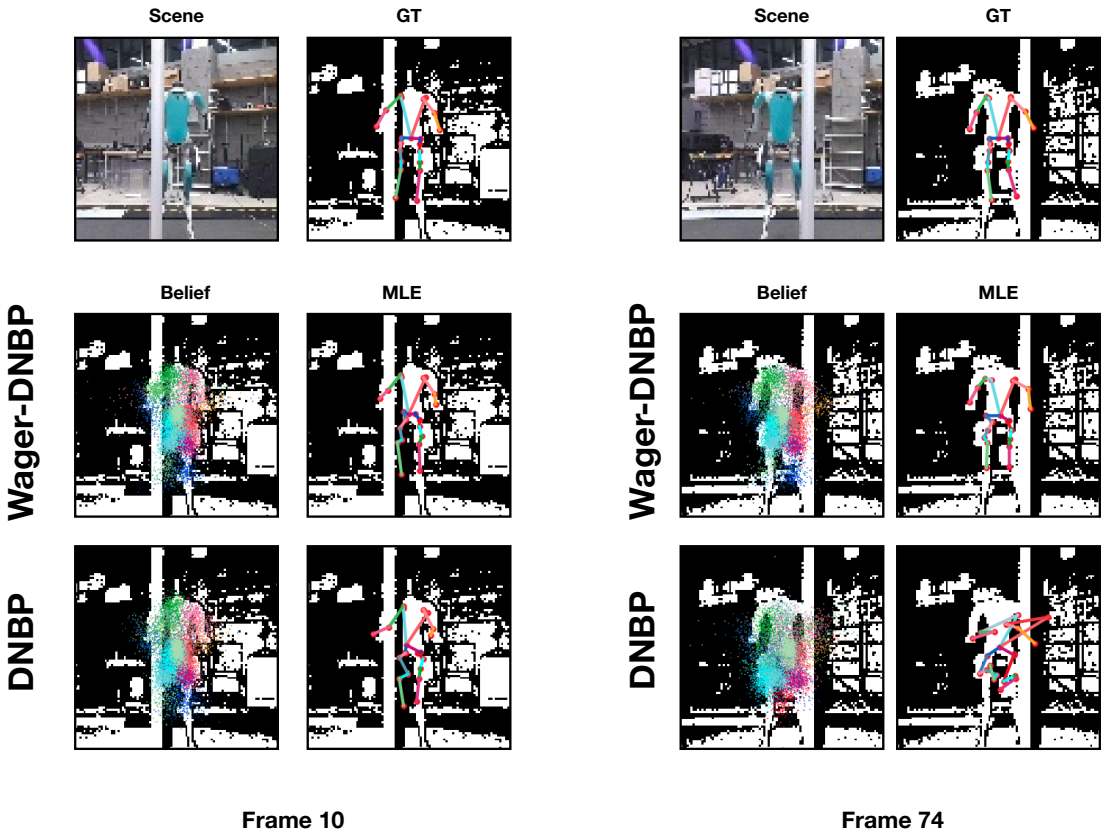


Figure 5.3: Qualitative results on a sequence of Digit laterally moving behind a pole at Frame 10 (left) and Frame 74 (right). While both methods perform well at the beginning of the movement, our method can maintain proper belief after passing behind the pole.

track of the robot. We see its visualization of the belief distribution become more sparse, and in turn, a visualization of its maximum likelihood estimate results in an implausible configuration for Digit. This analysis underscores the enhanced reliability of WAGER-DNBP in tracking scenarios.

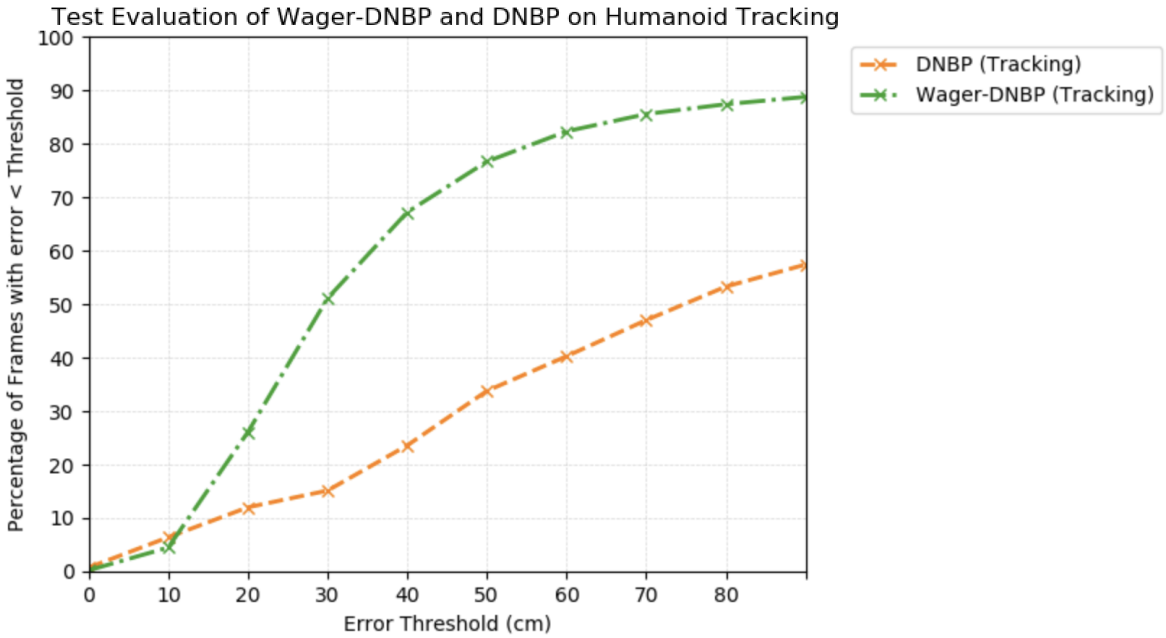


Figure 5.4: Quantitative results showing the percentage of estimates on the test dataset below error thresholds varying from $0 - 90cm$. DNBP [6] loses track of the robot more frequently, so it has a lower percentage of estimates below most thresholds.

We also present quantitative results over the test set in Figure 5.4. Both methods have similarly low performance for estimates with an error below $10cm$. However, Wager-DNBP has a significantly higher percentage of estimates that are within thresholds that are *generally* close. We observe that when either tracking estimate fails, the samples begin to disperse to far regions of the crop of the image. Therefore, many estimates are very wrong ($> 1m$).

5.6 Future Work

The quantitative results could be more meaningful to our objective of improving on occlusion through further analysis. We would need to distill the poses in the test set that were occluded. This could be done by comparing the measured depth of each joint against its known ground truth depth based on our annotations to determine if the joint is occluded. Additionally, we could use a segmentation network to estimate the mask of the visible portions of the robot in the image. These masks could be compared against the true segmentation from the rendering of our annotation to calculate the percentage of the robot visible. It would be interesting to look at how the methods perform across different levels of observability of the humanoid. Anecdotally, we observe the robot loses track when the amount of observability greatly changes between frames. By comparing the percentage of observability of the robot across frames, we could specifically look at performance relative to how observability changes.

5.7 Conclusions

In this work, we addressed the need for robots to operate effectively in complex environments, particularly focusing on their ability to track other entities using visual perception. We identified challenges with Differentiable Nonparametric Belief Propagation (DNBP), notably its computational constraints and potential ambiguities when analyzing specific viewpoints. To mitigate these issues, we integrated evidential reasoning with DNBP, leading to the development of the WAGER-DNBP method. This approach enhances nonparametric belief propagation by accounting for ambiguity and disbelief, especially in occlusion-rich real-world scenarios. Our results indicate that incorporating deep evidential reasoning into Bayesian graphical models allows for improved handling of ambiguities and better recovery from poor initializations.

This research contributes a robust method for improving robot navigation in dynamic environments.

Chapter 6

Conclusion and Future Directions

6.1 Conclusion

This dissertation introduces deep evidential reasoning as an informative signal to improve resampling in nonparametric Bayesian inference. We motivate the need for sampling-based filters to quantify the confidence and doubt associated with a sample and estimate the observation model's ability to measure these quantities given the observations at hand.

We introduce the concept of a counter-hypothetical likelihood function in our counter-hypothetical particle filter. This work posits that the standard practice of only quantifying supporting evidence is insufficient to infer doubt, as there is the presence of ambiguity in observations. It presents a formulation for how an independent measure of disqualifying evidence of a hypothesis could be integrated into a nonparametric Bayesian filter. We demonstrated how counter-hypothetical weightings for a hypothesis can indicate if the filter is in failure mode, showing improved performance. Its novelty comes from estimating doubt independently of likelihood or ambiguity and integrating these signals into nonparametric Bayesian inference.

We introduce a unique humanoid robot pose-tracking dataset to focus on the domain of tracking highly articulated objects of known models despite occlusion. In the Progress LUMBER Dataset, we can annotate 29 links of the humanoid robot,

Digit, with no external markers. While there are other pose-tracking datasets, they do not feature any external obstacles, so ours features a much wider variety of occlusions and ambiguous viewpoints. Additionally, we do not use a mounted or stationary manipulator but rather a walking bipedal robot, increasing the movement and difficulty of tracking.

Our work then extends the counter-hypothetical likelihood to a factor graph. It also shows how quantifying ambiguity can improve a factor graph's performance by estimating which observations contain the most useful information. We then demonstrate how this quantification of information in the observation can be integrated into the message-passing algorithm to adapt the influence the observation has on the samples' importance weightings. This work is novel for building on deep evidential reasoning to estimate ambiguity based on our independent estimates of likelihood and doubt. In this way, we do not require a dataset labeled for regions of ambiguity or occlusion.

6.2 Limitations

At a high level, our augmented likelihoods increase the time complexity of our inference, similar to ensemble methods. This burden is problematic for real-time robotics applications, especially when used as a workaround for more samples. Further analysis could be done on changes in performance and time constraints compared to increasing the number of samples. Additionally, further metrics and analysis are needed to fully understand the benefit of this addition in occluded test cases.

6.3 Future Directions

6.3.1 Data Collection: Clothing

While The Progress LUMBER Dataset is interesting, multiple supplementary sequences could be added to increase its usefulness. For example, human tracking works mention an inability to estimate pose when wearing baggy clothes, such as skirts. With Digit's human-like size, we could easily put clothes on the robot to showcase this edge case, and the robot's proprioceptive sensors could help with the accuracy of annotation for this task. Additionally, most of the dataset is front-facing, with side views and back only existing in 20% of the data, specifically in the turning sequences.

6.3.2 Counter-Hypothetical Unsupervised Learning

More work could be done to explore the generalizability and usefulness of the counter-hypothetical likelihood function. An area of interest is unsupervised learning, as real-world annotated perception datasets for robotics are limited. A standard likelihood function learning schema requires ground truth labeling. However, with the counter-hypothetical likelihood, it could use non-labeled data as a supervisory signal. For example, it can learn to produce high doubt weightings for estimates that are not physically plausible when simulated. Additionally, hypotheses that do not form trajectories consistently present in the image can also be labeled as incorrect.

6.3.3 Deep Evidential Reasoning

Dependencies and relationships within deep uncertainty quantification and evidential reasoning are not fully utilized or enforced. For example, we did not constrain Wager-DNBP to produce probability masses for likelihood, ambiguity, and doubt that sum to one. Adding this constraint could improve training performance. Additionally,

recognizing when these do not sum to one at test time could indicate another failure mode or edge case.

BIBLIOGRAPHY

[1] The parable of the kitchen counter: Maintaining cleanliness. https://goodmorningshelly.com/the-parable-of-the-kitchen-counter-maintaining-cleanliness/.

[2] Zhiqiang Sui, Zheming Zhou, Zhen Zeng, and Odest Chadwicke Jenkins. SUM: Sequential scene understanding and manipulation. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 3281–3288, 2017.

[3] Matias Valdenegro-Toro and Daniel Saromo Mori. A deeper look into aleatoric and epistemic uncertainty disentanglement. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1508–1516. IEEE, 2022.

[4] Michael Isard and Andrew Blake. Condensation—conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.

[5] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3d: A modern library for 3d data processing. *arXiv preprint arXiv:1801.09847*, 2018.

[6] Anthony Opipari, Jana Pavlasek, Chao Chen, Shoutian Wang, Karthik Desingh, and Odest Chadwicke Jenkins. Differentiable nonparametric belief propagation. *ICRA Workshop: Robotic Perception and Mapping: Emerging Techniques*, 2022.

[7] Nicolas Chopin, Omiros Papaspiliopoulos, et al. *An introduction to sequential Monte Carlo*, volume 4. Springer, 2020.

[8] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *ASME Journal of Fluids Engineering*, 1960.

[9] Judea Pearl. Chapter 4 - belief updating by network propagation. In Judea Pearl, editor, *Probabilistic Reasoning in Intelligent Systems*, pages 143 – 237. Morgan Kaufmann, San Francisco (CA), 1988.

[10] Erik B. Sudderth, Alexander T. Ihler, William T. Freeman, and Alan S. Willsky. Nonparametric belief propagation. In *Computer Vision and Pattern Recognition (CVPR)*, pages 605–612. IEEE Computer Society, 2003.

[11] Erik B Sudderth, Michael I Mandel, William T Freeman, and Alan S Willsky. Visual hand tracking using nonparametric belief propagation. In *Conference on Computer Vision and Pattern Recognition Workshop*, pages 189–189. IEEE, 2004.

[12] Leonid Sigal, Michael Isard, Benjamin Sigelman, and Michael Black. Attractive people: Assembling loose-limbed models using non-parametric belief propagation. *Advances in Neural Information Processing Systems*, 16, 2003.

[13] Jana Pavlasek, Stanley Lewis, Karthik Desingh, and Odest Chadwicke Jenkins. Parts-based articulated object localization in clutter using belief propagation. In *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020.

[14] Xinke Deng, Arsalan Mousavian, Yu Xiang, Fei Xia, Timothy Bretl, and Dieter Fox. PoseRBPF: A Rao-Blackwellized particle filter for 6D object pose tracking. In *Robotics: Science and Systems (RSS)*, 2019.

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

[16] Rico Jonschkowski, Divyam Rastogi, and Oliver Brock. Differentiable particle filters: End-to-end learning with algorithmic priors. In *Robotics: Science and Systems (RSS)*, 2018.

[17] Peter Karkus, David Hsu, and Wee Sun Lee. Particle filter networks with application to visual localization. In *Conference on Robot Learning (CoRL)*, pages 169–178. PMLR, 2018.

[18] Tishby, Levin, and Solla. Consistent inference of probabilities in layered networks: predictions and generalizations. In *International Joint Conference on Neural Networks 1989*, pages 403–409. IEEE, 1989.

[19] John Denker and Yann LeCun. Transforming neural-net output levels to probability distributions. *Advances in Neural Information Processing Systems*, 3, 1990.

[20] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[21] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, pages 1050–1059. PMLR, 2016.

[22] Dieter Fox. Kld-sampling: Adaptive particle filters. *Advances in Neural Information Processing Systems*, 14, 2001.

[23] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. In *Robotics: Science and Systems (RSS)*, 2018.

[24] Arthur P Dempster. A generalization of bayesian inference. *Journal of the Royal Statistical Society: Series B (Methodological)*, 30(2):205–232, 1968.

[25] Elizabeth A. Olson, Jana Pavlasek, Jasmine A. Berry, and Odest Chadwicke Jenkins. Counter-hypothetical particle filters for single object pose tracking. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3853–3859, 2023.

[26] Li Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, and Matti Pietikäinen. Deep learning for generic object detection: A survey. *International Journal of Computer Vision*, 128:261–318, 2020.

[27] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages I–I. Ieee, 2001.

[28] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893. Ieee, 2005.

[29] Pedro Felzenszwalb, David McAllester, and Deva Ramanan. A discriminatively trained, multiscale, deformable part model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. Ieee, 2008.

[30] Pedro F Felzenszwalb, Ross B Girshick, and David McAllester. Cascade object detection with deformable part models. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2241–2248. IEEE, 2010.

[31] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[32] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25, 2012.

[33] Christian Szegedy, Alexander Toshev, and Dumitru Erhan. Deep neural networks for object detection. *Advances in Neural Information Processing Systems*, 26, 2013.

[34] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.

[35] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.

[36] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015.

[37] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28, 2015.

[38] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *IEEE International Conference on Computer Vision*, pages 2961–2969, 2017.

[39] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88:303–338, 2010.

[40] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pages 740–755. Springer, 2014.

[41] Kai Chen and Qi Dou. Sgpa: Structure-guided prior adaptation for category-level 6d object pose estimation. In *IEEE/CVF International Conference on Computer Vision*, pages 2773–2782, 2021.

[42] Zhaoxin Fan, Zhengbo Song, Jian Xu, Zhicheng Wang, Kejian Wu, Hongyan Liu, and Jun He. Acr-pose: Adversarial canonical representation reconstruction network for category level 6d object pose estimation. *arXiv preprint arXiv:2111.10524*, 2021.

[43] Meng Tian, Marcelo H Ang, and Gim Hee Lee. Shape prior deformation for categorical 6d object pose and size estimation. In *European Conference on Computer Vision (ECCV)*, pages 530–546. Springer, 2020.

[44] Muhammad Zubair Irshad, Thomas Kollar, Michael Laskey, Kevin Stone, and Zsolt Kira. Centersnap: Single-shot multi-object 3d shape reconstruction and categorical 6d pose and size estimation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 10632–10640. IEEE, 2022.

[45] Jiehong Lin, Zewei Wei, Zhihao Li, Songcen Xu, Kui Jia, and Yuanqing Li. Dualposenet: Category-level 6d object pose and size estimation using dual pose network with refined learning of pose consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3560–3569, 2021.

[46] Xinke Deng, Junyi Geng, Timothy Bretl, Yu Xiang, and Dieter Fox. iCaps: Iterative category-level object pose and shape estimation. *Robotics and Automation Letters (RA-L)*, 2022.

[47] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2642–2651, 2019.

[48] Bertram Drost, Markus Ulrich, Nassir Navab, and Slobodan Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 998–1005. Ieee, 2010.

[49] Joel Vidal, Chyi-Yeu Lin, and Robert Martí. 6d pose estimation using an improved method based on point pair features. In *International Conference on Control, Automation and Robotics (ICCAR)*, pages 405–409. IEEE, 2018.

[50] Radu Bogdan Rusu, Gary Bradski, Romain Thibaux, and John Hsu. Fast 3d recognition and pose using the viewpoint feature histogram. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2155–2162. IEEE, 2010.

[51] Zoltan-Csaba Marton, Dejan Pangercic, Nico Blodow, and Michael Beetz. Combined 2d–3d categorization and classification for multimodal perception systems. *The International Journal of Robotics Research*, 30(11):1378–1402, 2011.

[52] Pauline C Ng and Steven Henikoff. Sift: Predicting amino acid changes that affect protein function. *Nucleic Acids Research*, 31(13):3812–3814, 2003.

[53] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European Conference on Computer Vision (ECCV)*, pages 404–417. Springer, 2006.

[54] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *International Conference on Computer Vision ICCV)*, pages 2564–2571. Ieee, 2011.

[55] Tomáš Hodaň, Xenophon Zabulis, Manolis Lourakis, Štěpán Obdržálek, and Jiří Matas. Detection and fine 3d pose estimation of texture-less objects in rgb-d images. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4421–4428. IEEE, 2015.

[56] Stefan Hinterstoisser, Stefan Holzer, Cedric Cagniart, Slobodan Ilic, Kurt Konolige, Nassir Navab, and Vincent Lepetit. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In *International Conference on Computer Vision (ICCV)*, pages 858–865. IEEE, 2011.

[57] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4561–4570, 2019.

[58] Mahdi Rad and Vincent Lepetit. Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 3828–3836, 2017.

[59] Georgios Pavlakos, Xiaowei Zhou, Aaron Chan, Konstantinos G Derpanis, and Kostas Daniilidis. 6-dof object pose from semantic keypoints. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2011–2018. IEEE, 2017.

[60] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. *arXiv preprint arXiv:2306.13643*, 2023.

[61] Jonathan Tremblay, Thang To, Balakumar Sundaralingam, Yu Xiang, Dieter Fox, and Stan Birchfield. Deep object pose estimation for semantic robotic grasping of household objects. In *Conference on Robot Learning (CoRL)*, 2018.

[62] Jia Kang, Wenjun Liu, Wenzhe Tu, and Lu Yang. Yolo-6d+: single shot 6d pose estimation using privileged silhouette information. In *2020 International Conference on Image Processing and Robotics (ICIP)*, pages 1–6. IEEE, 2020.

[63] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, 2016.

[64] Yinlin Hu, Joachim Hugonot, Pascal Fua, and Mathieu Salzmann. Segmentation-driven 6d object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3385–3394, 2019.

[65] Yongzhi Su, Mahdi Saleh, Torben Fetzer, Jason Rambach, Nassir Navab, Benjamin Busam, Didier Stricker, and Federico Tombari. Zebrapose: Coarse to fine surface encoding for 6dof object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6738–6748, 2022.

[66] Lin Yen-Chen, Pete Florence, Jonathan T Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. inerf: Inverting neural radiance fields for pose estimation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1323–1330. IEEE, 2021.

[67] Zheming Zhou, Xiaotong Chen, and Odest Chadwicke Jenkins. Lit: Light-field inference of transparency for refractive object localization. *IEEE Robotics and Automation Letters*, 5(3):4548–4555, 2020.

[68] Xiaotong Chen, Huijie Zhang, Zeren Yu, Anthony Opipari, and Odest Chadwicke Jenkins. Clearpose: Large-scale transparent object dataset and benchmark. In *European Conference on Computer Vision (ECCV)*, pages 381–396. Springer, 2022.

[69] Huijie Zhang, Anthony Opipari, Xiaotong Chen, Jiyue Zhu, Zeren Yu, and Odest Chadwicke Jenkins. Transnet: Category-level transparent object pose estimation. In *European Conference on Computer Vision*, pages 148–164. Springer, 2022.

[70] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. Deepim: Deep iterative matching for 6d pose estimation. In *European Conference on Computer Vision (ECCV)*, pages 683–698, 2018.

[71] Yann Labbé, Justin Carpentier, Mathieu Aubry, and Josef Sivic. CosyPose: Consistent multi-view multi-object 6d pose estimation. In *European Conference on Computer Vision*, pages 574–591. Springer, 2020.

[72] Bowen Wen, Chaitanya Mitash, Baozhang Ren, and Kostas E Bekris. se(3)-TrackNet: Data-driven 6D pose tracking by calibrating image residuals in synthetic domains. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 10367–10373. IEEE, 2020.

[73] Yann Labbé, Justin Carpentier, Mathieu Aubry, and Josef Sivic. Single-view robot pose and joint angle estimation via render & compare. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1654–1663, 2021.

[74] Nick Heppert, Toki Migimatsu, Brent Yi, Claire Chen, and Jeannette Bohg. Category-independent articulated object tracking with factor graphs. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3800–3807. IEEE, 2022.

[75] Neil J Gordon, David J Salmond, and Adrian FM Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE Proceedings F (Radar and Signal Processing)*, volume 140, pages 107–113. IET, 1993.

[76] Genshiro Kitagawa. Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25, 1996.

[77] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte carlo localization for mobile robots. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, volume 2, pages 1322–1328 vol.2, 1999.

[78] Kevin Murphy, Yair Weiss, and Michael I Jordan. Loopy belief propagation for approximate inference: An empirical study. *arXiv preprint arXiv:1301.6725*, 2013.

[79] Michael Isard. PAMPAS: Real-valued graphical models for computer vision. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, 2003.

[80] Sebastian Reich. A nonparametric ensemble transform method for bayesian inference. *SIAM Journal on Scientific Computing*, 35(4):A2013–A2024, 2013.

[81] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in Neural Information Processing Systems*, 26, 2013.

[82] Adrien Corenflos, James Thornton, George Deligiannidis, and Arnaud Doucet. Differentiable particle filtering via entropy-regularized optimal transport. In *International Conference on Machine Learning*, pages 2100–2111. PMLR, 2021.

[83] Jonathan Deutscher, Andrew Blake, and Ian Reid. Articulated body motion capture by annealed particle filtering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 126–133. IEEE, 2000.

[84] S. Lenser and M. Veloso. Sensor resetting localization for poorly modelled mobile robots. In *International Conference on Robotics and Automation (ICRA)*, volume 2, pages 1225–1232, 2000.

[85] J-S Gutmann and Dieter Fox. An experimental comparison of localization methods continued. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 454–459. IEEE, 2002.

[86] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. MIT Press, 2005.

[87] Naoki Akai, Takatsugu Hirayama, and Hiroshi Murase. Hybrid localization using model-and learning-based methods: Fusion of monte carlo and e2e localizations via importance sampling. In *International Conference on Robotics and Automation (ICRA)*, pages 6469–6475. IEEE, 2020.

[88] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on Fairness, Accountability and Transparency*, pages 77–91. PMLR, 2018.

[89] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

[90] Zhen Zeng, Yunwen Zhou, Odest Chadwicke Jenkins, and Karthik Desingh. Semantic mapping with simultaneous object detection and localization. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 911–918. IEEE, 2018.

[91] David Gunning. Explainable artificial intelligence (xai). *Defense Advanced Research Projects Agency (DARPA), and Web*, 2(2):1, 2017.

[92] Dustin Tran, Jeremiah Liu, Michael W Dusenberry, Du Phan, Mark Collier, Jie Ren, Kehang Han, Zi Wang, Zelda Mariet, Huiyi Hu, et al. Plex: Towards reliability using pretrained large model extensions. *arXiv preprint arXiv:2207.07411*, 2022.

[93] Joris Guérin, Kevin Delmas, Raul Ferreira, and Jérémie Guiochet. Out-of-distribution detection is not all you need. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 14829–14837, 2023.

[94] Craig R. Fox and Gulden Ulkumen. Distinguishing two dimensions of uncertainty. *SSRN Electronic Journal*, 2011.

[95] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in Neural Information Processing Systems*, 30, 2017.

[96] Wenchong He and Zhe Jiang. A survey on uncertainty quantification methods for deep neural networks: An uncertainty source perspective. *arXiv preprint arXiv:2302.13425*, 2023.

[97] Stefan Depeweg, Jose-Miguel Hernandez-Lobato, Finale Doshi-Velez, and Steffen Udluft. Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning. In *International Conference on Machine Learning*, pages 1184–1193. PMLR, 2018.

[98] Wray L Buntine. Bayesian backpropagation. *Complex systems*, 5:603–643, 1991.

[99] Jouko Lampinen and Aki Vehtari. Bayesian approach for neural networks—review and case studies. *Neural Networks*, 14(3):257–274, 2001.

[100] Ethan Goan and Clinton Fookes. Bayesian neural networks: An introduction and survey. *Case Studies in Applied Bayesian Data Science: CIRM Jean-Morlet Chair, Fall 2018*, pages 45–87, 2020.

[101] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in Neural Information Processing Systems*, 30, 2017.

[102] Guanya Shi, Yifeng Zhu, Jonathan Tremblay, Stan Birchfield, Fabio Ramos, Animashree Anandkumar, and Yuke Zhu. Fast uncertainty quantification for deep object pose estimation. In *International Conference on Robotics and Automation (ICRA)*. IEEE, 11 2021.

[103] Maithra Raghu, Katy Blumer, Rory Sayres, Ziad Obermeyer, Bobby Kleinberg, Sendhil Mullainathan, and Jon Kleinberg. Direct uncertainty prediction for medical second opinions. In *International Conference on Machine Learning*, pages 5281–5290. PMLR, 2019.

[104] Tiago Ramalho and Miguel Miranda. Density estimation in representation space to predict model uncertainty. In *Engineering Dependable and Secure Machine Learning Systems: Third International Workshop, EDSMLS 2020, New York City, NY, USA, February 7, 2020, Revised Selected Papers 3*, pages 84–96. Springer, 2020.

[105] Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10951–10960, 2020.

[106] Glenn Shafer. *A mathematical theory of evidence*, volume 42. Princeton University Press, 1976.

[107] Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan kaufmann, 1988.

[108] Murat Sensoy, Lance Kaplan, and Melih Kandemir. Evidential deep learning to quantify classification uncertainty. *Advances in Neural Information Processing Systems*, 31, 2018.

[109] Andrey Malinin and Mark Gales. Predictive uncertainty estimation via prior networks. *Advances in Neural Information Processing Systems*, 31, 2018.

[110] Taejong Joo, Uijung Chung, and Min-Gwan Seo. Being bayesian about categorical probability. In *International Conference on Machine Learning*, pages 4950–4961. PMLR, 2020.

[111] Alexander Amini, Wilko Schwarting, Ava Soleimany, and Daniela Rus. Deep evidential regression. *Advances in Neural Information Processing Systems*, 33:14927–14937, 2020.

[112] Elizabeth A Olson, Jana Pavlasek, Jasmine A Berry, and Odest Chadwicke Jenkins. Counter-hypothetical particle filters for single object pose tracking. *arXiv preprint arXiv:2305.17828*, 2023.

[113] Manuel Wuthrich, Peter Pastor, Mrinal Kalakrishnan, Jeannette Bohg, and Stefan Schaal. Probabilistic object tracking using a range camera. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 3195–3202. IEEE, 2013.

[114] Zhiqiang Sui, Odest Chadwicke Jenkins, and Karthik Desingh. Axiomatic particle filtering for goal-directed robotic manipulation. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 4429–4436. IEEE, 2015.

[115] Dieter Fox, Wolfram Burgard, Frank Dellaert, and Sebastian Thrun. Monte carlo localization: Efficient position estimation for mobile robots. *AAAI/IAAI*, 1999(343-349):2–2, 1999.

[116] Dieter Fox. Adapting the sample size in particle filters through kld-sampling. *The International Journal of Robotics Research*, 22(12):985–1003, 2003.

[117] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martín-Martín, Cewu Lu, Li Fei-Fei, and Silvio Savarese. DenseFusion: 6D object pose estimation by iterative dense fusion. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3343–3352, 2019.

[118] Yisheng He, Haibin Huang, Haoqiang Fan, Qifeng Chen, and Jian Sun. FFB6D: A full flow bidirectional fusion network for 6D pose estimation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3003–3013, 2021.

[119] Manuel Stoiber, Martin Sundermeyer, and Rudolph Triebel. Iterative corresponding geometry: Fusing region and depth for highly efficient 3D tracking of textureless objects. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6855–6865. IEEE/CVF, 2022.

[120] Shile Li, Seongyong Koo, and Dongheui Lee. Real-time and model-free object tracking using particle filter with joint color-spatial descriptor. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 6079–6085. IEEE, 2015.

[121] Changhyun Choi and Henrik I Christensen. Robust 3d visual tracking using particle filtering on the special euclidean group: A combined approach of keypoint and edge features. *The International Journal of Robotics Research*, 31(4):498–519, 2012.

[122] Karthik Desingh, Odest Chadwicke Jenkins, Lionel Reveret, and Zhiqiang Sui. Physically plausible scene estimation for manipulation in clutter. In *International Conference on Humanoid Robots (Humanoids)*, pages 1073–1080. IEEE, 2016.

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

[124] Zhen Zeng, Adrian Röfer, and Odest Chadwicke Jenkins. Semantic linking maps for active visual object search. In *International Conference on Robotics and Automation (ICRA)*. IEEE, 2020.

[125] Alexander Ihler and David McAllester. Particle belief propagation. In *Artificial intelligence and statistics*, pages 256–263. PMLR, 2009.

[126] Karthik Desingh, Shiyang Lu, Anthony Opipari, and Odest Chadwicke Jenkins. Efficient nonparametric belief propagation for pose estimation and manipulation of articulated objects. *Science Robotics*, 4(30), 2019.

[127] Patrick Pfaff, Wolfram Burgard, and Dieter Fox. Robust Monte-Carlo localization using adaptive likelihood models. In *European robotics symposium 2006*, pages 181–194. Springer, 2006.

[128] Sebastian Thrun, Dieter Fox, Wolfram Burgard, and Frank Dellaert. Robust monte carlo localization for mobile robots. *Artificial Intelligence*, 128(1-2):99–141, 2001.

[129] Lei Zhang, Rene Zapata, and Pascal Lepinay. Self-adaptive Monte Carlo localization for mobile robots using range finders. *Robotica*, 30(2):229–244, 2012.

[130] Joseph Y Halpern. *Reasoning about uncertainty*. MIT Press, 2017.

[131] Agility Robotics. Digit robot. https://agilityrobotics.com/robots.

[132] Timothy E Lee, Jonathan Tremblay, Thang To, Jia Cheng, Terry Mosier, Oliver Kroemer, Dieter Fox, and Stan Birchfield. Camera-to-robot pose estimation from a single image. In *International Conference on Robotics and Automation (ICRA)*, 2020.

[133] Yiming Zuo, Weichao Qiu, Lingxi Xie, Fangwei Zhong, Yizhou Wang, and Alan L Yuille. Craves: Controlling robotic arm with a vision-based economic system. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4214–4223, 2019.

[134] Manuel Stoiber, Martin Sundermeyer, Wout Boerdijk, and Rudolph Triebel. A multi-body tracking framework–from rigid objects to kinematic structures. *arXiv preprint arXiv:2208.01502*, 2022.

[135] P.J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[136] Dmitry Chetverikov, Dmitry Svirko, Dmitry Stepanov, and Pavel Krsek. The trimmed iterative closest point algorithm. In *2002 International Conference on Pattern Recognition*, volume 3, pages 545–548. IEEE, 2002.

[137] Ken Shoemake. Animating rotation with quaternion curves. In *Computer Graphics and Interactive Techniques*, pages 245–254, 1985.

[138] Jakob Gawlikowski, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, et al. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, pages 1–77, 2023.

[139] Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U Rajendra Acharya, et al. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, 76:243–297, 2021.

[140] Hugo Grimmett, Rudolph Triebel, Rohan Paul, and Ingmar Posner. Introspective classification for robot perception. *The International Journal of Robotics Research*, 35(7):743–762, 2016.

[141] Tom Williams, Fereshta Yazdani, Prasanth Suresh, Matthias Scheutz, and Michael Beetz. Dempster-shafer theoretic resolution of referential ambiguity. *Autonomous Robots*, 43:389–414, 2019.

[142] Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning*, 110:457–506, 2021.

[143] Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you

trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in Neural Information Processing Systems*, 32, 2019.

[144] Andrew G Wilson and Pavel Izmailov. Bayesian deep learning and a probabilistic perspective of generalization. *Advances in Neural Information Processing Systems*, 33:4697–4708, 2020.

[145] Karthik Desingh, Shiyang Lu, Anthony Opipari, and Odest Chadwicke Jenkins. Factored pose estimation of articulated objects using efficient nonparametric belief propagation. In *International Conference on Robotics and Automation (ICRA)*, pages 7221–7227, 2019.

[146] Jason Pacheco, Silvia Zuffi, Michael Black, and Erik Sudderth. Preserving modes and messages via diverse particle selection. In *International Conference on Machine Learning*, pages 1152–1160. PMLR, 2014.

[147] Jana Pavlasek, Stanley Lewis, Karthik Desingh, and Odest Chadwicke Jenkins. Parts-based articulated object localization in clutter using belief propagation. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10595–10602, 2020.

[148] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.