Data Association for Semantic World Modeling from Partial Views

Lawson L.S. Wong, Leslie Pack Kaelbling, Tomás Lozano-Pérez
CSAIL, MIT, Cambridge, MA 02139
{ lsw, lpk, tlp }@csail.mit.edu

Abstract

Autonomous mobile-manipulation robots need to sense and interact with objects to accomplish high-level tasks such as preparing meals and searching for objects. To achieve such tasks, robots need semantic world models, defined as object-based representations of the world involving task-level attributes. In this work, we address the problem of estimating world models from semantic perception modules that provide noisy observations of attributes. Because attribute detections are sparse, ambiguous, and are aggregated across different viewpoints, it is unclear which attribute measurements are produced by the same object, so *data association* issues are prevalent. We present novel clustering-based approaches to this problem, which are more efficient and require less severe approximations compared to existing tracking-based approaches. These approaches are applied to data containing object type-and-pose detections from multiple viewpoints, and demonstrate comparable quality using a fraction of the computation time.

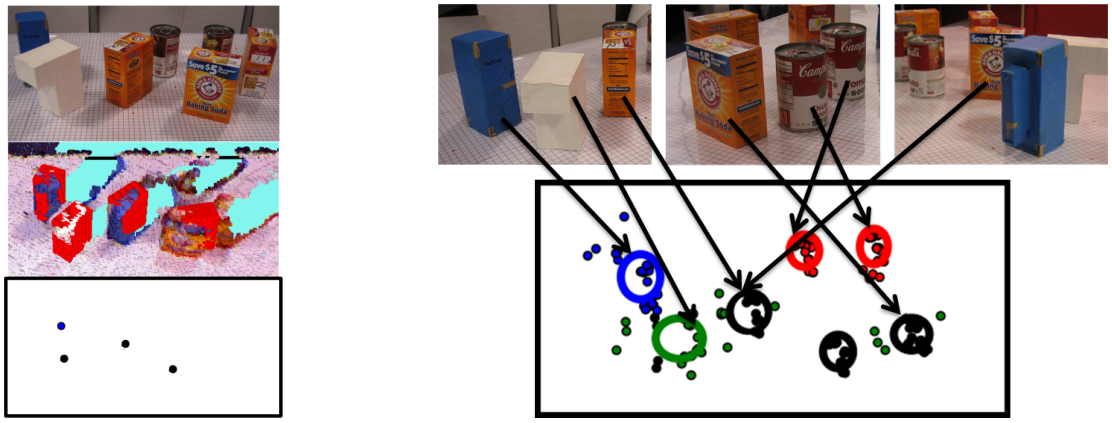
1 Introduction

Much of the everyday human physical environment is made up of coherent physical objects. Environmental dynamics are well described in terms of the effects of actions on those objects. Perceptual systems are able to report detections of objects with type, location, color, and other properties. Humans naturally designate both goals and prior information in terms of objects. Thus, it is appropriate for robots to construct 'mental models' of their environment that are structured around objects, their properties, and their relations to one another.

In this work, we define a semantic world model to be a set of objects with associated attributes and relations. To illustrate this concept concretely, consider the following tasks, along with objects and attributes that are potentially relevant:

- Cooking eggs on a pan: Objects — Eggs, pan, stove, etc.
  Attributes — *CookedTime*, *StoveSetting*, *EggPositionRelativeToPan*
- Finding chairs for guests: Objects — Furniture, people
  Attributes — *IsChair*, *Sittable*, *Movable*, *Location*, *SittingOn(Person, Furniture)*
- Rearranging objects on a table: Objects — Items on table
  Attributes — *Shape*, *Type*, *RelativePositionAndOrientation*, *GraspPoints*

A common theme underlying these tasks, and many others, is that successful planning and execution hinges on good world-state estimation and monitoring. Dynamic attributes listed above also highlight why object-based representations are uniquely suitable for dynamic tasks: transition



(a) Single viewpoint **(b)** Aggregation of object detections from multiple viewpoints

Figure 1: (a) Given a tabletop scene (top), we want to estimate the types and poses of objects in the scene using a black-box object detector. From a single RGB-D image, however, objects may be occluded or erroneously classified. In the rendered image (middle; detections superimposed in red), three objects are missing due to occlusion, and the bottom two objects have been misidentified. The semantic attributes that result in our representation are very sparse (bottom; dot location is measured 2-D pose, color represents type). **(b)** Aggregation of measurements from many different viewpoints (top) is therefore needed to construct good estimates. However, this introduces data association issues of the type addressed in this work, especially when multiple instances of the same object type are present. From all the object detection data, as shown (bottom) by dots (each dot is one detection), our goal is to estimate the object types and poses in the scene (shown as thick ellipses centered around location estimate; color represents type, ellipse size reflects uncertainty). The estimate above identifies all types correctly with minimal error in pose.

dynamics tends to operate on the level of objects. For example, it is much more natural to express and reason about eggs that are being cooked, as opposed to points in a point cloud or cells in an occupancy grid that are 'cooked'. Although we focus on the static case in this paper, our ultimate goal is to provide a framework for estimating and monitoring large semantic world models involving objects and attributes that change over time as a result of physical processes as well as actions by the robot and other agents.

In this work, we address the problem of constructing world models from semantic perception modules that provide noisy observations of attributes. For concreteness, Figure 1 depicts an application of our methods; here the world model consists of objects' types and poses, and attribute measurements are outputs from a black-box object detector running continuously on sensed RGB-D images. Due to noise, occlusion, and sensors' limited field of view, observations from multiple viewpoints will typically be necessary to produce a confident world model. Because attribute detections are sparse, noisy, and inherently ambiguous, where it is unclear which attribute measurements were produced by the same object across different views, *data association* issues become critical. This is the greatest challenge; if the measurement-object correspondences were known, the resulting object-attribute posterior distributions would be efficiently computable.

We begin by stating a formal model for a simplified 1-D version of the world-model estimation problem in Section 3, and then review a classic solution approach based on tracking in Section 4. The main contribution of this work is the development of several novel clustering-based data association approaches, described in Sections 5 and 6. Application of the semantic world-modeling framework to object type-and-pose estimation is then demonstrated in Section 7, where we present experimental results using data collected with a Kinect sensor on a mobile robot.

2 Related Work

Our work lies in the intersection of semantic perception, world modeling, and data association, which we will first review before placing our contributions in context.

2.1 Semantic World Modeling

Understanding the mobile robot's spatial environment, by deriving a world model from its sensors, has long been a problem of interest to the robotics community (Crowley, 1985). Early work typically focused on using ultrasonic range sensors, tracking low-level planar and corner features as landmarks in a map (Cox and Leonard, 1994). The resulting geometric maps were useful for mobile robot navigation, but objects were not a primary concern in these representations and tasks.

For mobile-manipulation robots that operate on objects, the world model must contain information about objects in the world. With the advent of more effective visual sensors, image features, and object detectors, world models are now capable of supporting richer representations of objects. The important role of objects in spatial representations was explored by Ranganathan and Dellaert (2007), where places were modeled using objects as the basic unit of representation. However, like much of the related work in semantic mapping (Vasudevan et al., 2007; Zender et al., 2008; Nüchter and Hertzberg, 2008; Pronobis and Jensfelt, 2012), the ultimate goal is place modeling and recognition, which is most useful for navigation. Instead, we want to infer the precise object states themselves, which are needed for mobile-manipulation tasks.

To measure object states, we rely on attribute detectors, particularly ones operating on 3-D visual data. Object recognition and pose estimation has received widespread attention from the computer vision and robotics communities. With the recent advances in RGB-D cameras, several systems have been developed to detect object types/instances and their 6-D poses from 3-D point clouds (Rusu et al., 2010; Glover et al., 2011; Lai et al., 2012; Aldoma et al., 2013; Marton et al., 2014). We will use one such detector (Glover and Popovic, 2013) as our black-box attribute detector, but we emphasize that our methods are agnostic to the detector used.

A basic world model could simply use a detector's output on a single image as a representation of the world. However, this suffers from many sources of error: sensor measurement noise, object occlusion, and modeling and approximation errors in the detection algorithms. As motivated in the previous section, aggregating measurements across different viewpoints can help reduce estimation error. For example, Hager and Wegbreit (2011) demonstrate the utility of considering a prior 3-D scene model and its potential evolution over scenes. Using this observation as a premise, active perception approaches (e.g., Eidenberger and Scharinger (2010); Velez et al. (2012); Atanasov et al. (2013)) seek the next best view (camera pose) where previously-occluded objects may be visible, typically by formulating the problem as a partially-observable Markov decision process. Because the focus is on planning instead of estimation, this line of work is complementary to the world modeling problem, which considers estimation using measurements from an uncontrolled, arbitrary collection of camera poses.

The primary challenge in aggregating object detections across multiple views of the world is identity management, induced by the fact that measurements often cannot uniquely mapped to an underlying object in the world. Blodow et al. (2010) formulated object identity resolution as an inference problem in a Markov logic network, but acknowledge the complexity of their approach. Most similar to our approach is the work of Elfring et al. (2013), which highlighted the data association issues in semantic world modeling, and applied a classic multiple hypothesis tracking

(MHT) approach to the problem. The limitations of MHTs will be discussed in the next subsection, in the context of other data association methods, and revisited in Section 4.

Recently, besides estimating object states in the world via object attribute detections, there has been interest in world modeling involving object information, but without explicit recognition. As mentioned above, this is often the case for semantic mapping. Anati et al. (2012) showed that object-based robot localization is still possible even if "soft" heatmaps of local image features are used instead of explicit object poses. Mason and Marthi (2012) argue that, for long-term and large-scale mapping, modeling and recognizing all objects is impractical. The recent success of dense 3-D reconstruction (Newcombe et al., 2011) has also led to dense surface maps being a viable representation of space. Discussion of which representation is the best for world modeling is beyond the scope of this paper, and depends on the considered domain/task. Moreover, we emphasize that object type-and-pose estimation was only chosen as a concrete and familiar proof-of-concept application. Most of the presented related work is specific to this application, whereas our framework is applicable to other semantic attributes and tasks.

2.2 Data Association

The data association problem was historically motivated by target tracking; Bar-Shalom and Fortmann (1988) provide a comprehensive overview of the foundations, as well as coverage of greedy nearest-neighbor methods and an approximate Bayesian filter, the joint probabilistic data association filter (JPDAF). Apart from being a suboptimal approximation, the JPDAF is also limited by its assumption of a fixed number of tracked targets (objects), which is not valid for our problem.

A more principled approach when the number of tracks is unknown is multiple hypothesis tracking (MHT) (Reid, 1979). In principle, MHT considers the tree of all possible association hypotheses, branching on the possible tracks that each measurement can correspond to. However, due to the number of measurements involved, maintaining the entire tree (and hence the exact posterior distribution) is exponentially expensive and intractable for any non-trivial branching factor. As a result, practical implementations of MHTs must use one of many proposed heuristics (e.g., Kurien (1990); Cox and Hingorani (1996)), typically pruning away all but the few most-likely branches in the association tree. Aggressive pruning potentially removes correct associations that happen to appear unlikely at the moment. Although this problem is somewhat mitigated by postponing ambiguous associations through delayed filtering, the window for resolving issues is short because of computational limitations.

The MHT pruning heuristics were necessitated by the combinatorial complexity of MHT, which in turn is due to the enumeration of all possible association histories. Instead of attempting to evaluate every point in this large space, most of which contains little probability mass, efficient sampling techniques have been proposed that try to only explore high-probability regions. Markov-chain Monte Carlo (MCMC) methods for sampling association matchings and tracks have been explored by Dellaert et al. (2003) for structure-from-motion and by Pasula et al. (1999) for traffic surveillance. More recently, Oh et al. (2009) generalized the latter work by considering a wider class of transition moves during sampling, and provided theoretical bounds on the mixing (convergence) time of their sampling algorithm, MCMCDA. Because only a small space of likely associations is frequently sampled, and all measurement associations are repeatedly considered (unlike MHT with pruning), MCMCDA empirically outperforms MHT both in efficiency and accuracy, especially in environments with heavy detection noise.

Apart from the advantages of MCMC sampling methods, Dellaert (2001) also recognized the

utility of considering attributes in data association problems. When occlusion and clutter are present, correspondences are frequently ambiguous, and incorporating more information can help separate the targets of correspondence. Dellaert (2001) specifically considered this idea in the context of the structure-from-motion problem, proposing that image feature appearances should be considered in addition to their measured locations, in order to better distinguish different features between images. Our approach shares many resemblances to this line of work due to the use of attributes and sampling-based inference.

2.3 Contributions

In the context of previous work, we view our approach as building on the semantic world modeling problem formulation of Elfring et al. (2013) and the data association techniques of Oh et al. (2009). As argued above and by Oh et al. (2009), MHT has various drawbacks, which are directly inherited by the approach of Elfring et al. (2013). However, instead of directly applying MCMCDA to world modeling, we will introduce more domain assumptions to make inference more efficient.

Unlike target tracking, for which most data association algorithms are designed, semantic world modeling has three distinguishing domain characteristics:

- Objects can have attributes besides location, and hence are distinguishable from each other in general (which likely makes data association easier). Some data association methods can be readily generalized to this case (as was done by Elfring et al. (2013)), but it excludes some from consideration, such as the probability hypothesis density (PHD) filter by Mahler (2007).
- Only a small region of the world is visible from any viewpoint. Most data association methods operate in regimes where all targets are sensed (possibly with noise/failure) at each time point.
- Most object states do not change over short periods of time.

In light of the final point, we study the semantic world modeling problem under the stringent assumption that the world is static, i.e., object states do not change.[1] This does not trivialize the data association problem, since it is still necessary to determine measurement-to-object correspondences (and is exacerbated by the limited field of view). However, target-tracking algorithms no longer seem most appropriate, since time is no longer an essential dimension. Instead, the problem becomes more akin to *clustering*, where objects are represented by points in the joint attribute (product) space, and measurements form clusters around these points.

A useful model for performing clustering with an unbounded number of clusters is the Dirichlet process mixture model (DPMM) (Antoniak, 1974; Neal, 2000), a Bayesian nonparametric approach that can be viewed as an elegant extension to finite mixture models. We apply this method to world modeling in Section 5 and derive a Gibbs sampling algorithm to perform inference. The sampling candidate proposals in this algorithm can be viewed as a subset of those considered by Oh et al. (2009). However, clustering ignores a crucial assumption in data association; more details will be given in Section 6, where we also introduce modifications and approximations to address this issue.

[1] Over long periods of time, this assumption is clearly unrealistic, but is beyond the scope of this paper. A naïve solution is to refresh the world model using a short window of measurements prior to each query, assuming that the world has not changed during that window.

3 The 1-D Colored-Lights Domain

For clarity of explanation we begin by introducing a model of minimal complexity, involving objects with 1-D locations and a single attribute (color). Despite this simplification, the fundamental issues in data association are captured in the model described in this section. Generalizing to higher dimensions and more attributes is relatively straightforward; in Section 7, we generalize to 3-D locations and use object types as an attribute in our semantic world modeling application.

The world consists of an unknown number (K) of stationary lights. Each light is characterized by its color c_k and its location $l_k \in \mathbb{R}$, both of which do not change over time. A finite universe of colors of size C is assumed. A robot moves along this 1-D world, occasionally gathering partial views of the world with known fields of view $[a^v, b^v] \subset \mathbb{R}$. Within each view, M^v lights of various colors and locations are observed, denoted by $o_m^v \in [C] \triangleq \{1, \ldots, C\}$ and $x_m^v \in \mathbb{R}$ respectively. These (o_m^v, x_m^v) pairs may be noisy (in both color and location) or spurious (false positive – FP) measurements of the true lights. Also, a light may sometimes fail to be perceived (false negative – FN). Given these measurements, the goal is to determine the posterior distribution over configurations (number, colors, and locations) of lights in the explored region of the world.

We assume the following form of noise models. For color observations, for each color c, there is a known discrete distribution $\phi^c \in \Delta^C$ (estimable from perception apparatus statistics) specifying the probability of color observations:

$$\phi_i^c = \begin{cases} \mathbb{P}(\text{no observation for light with color } c)\ , & i = 0 \\ \mathbb{P}(\text{color } i \text{ observed for light with color } c)\ , & i \in [C] \end{cases} \tag{1}$$

A similar distribution ϕ^0 specifies the probability of observing each color given that the observation was a false positive. False positives are assumed to occur in a proportion p_{FP} of object detections. Each view may have multiple detections and hence multiple false positives. For location observations, if the observation corresponds to an actual light, then the observed location is assumed to be Gaussian-distributed, centered on the actual location. The variance is *not* assumed known and will be estimated for each light from measurement data. For false positives, the location is assumed to be uniformly distributed over the field of view (Unif$[a^v, b^v]$).

Next, we present the core problem of this domain. Given sets of color-location detections from a sequence of views, $\{\{(o_m^v, x_m^v)\}_{m=1}^{M^v}\}_{v=1}^{V}$, we want to infer the posterior distribution on the configuration of lights $\{(c_k, l_k)\}_{k=1}^{K}$, where K is unknown as well. If we knew, for each light, which subset of the measurements were generated from that light, then we would get K decoupled estimation problems (assuming lights are independent from each other). With suitable priors, these single-light estimation problems admit efficient solutions; details can be found in the Appendix.

The issue is that these associations are unknown. Therefore, we must reason over the *space* of possible data associations. For each observation, let z_m^v be the index of the light that the observation corresponds to (ranging in $[K]$ for a configuration with K lights), or 0 if the observation is a false positive. z_m^v is the latent association for measurement (o_m^v, x_m^v). Let $\mathbf{z}^v$ be the concatenated length-M^v vector of all z_m^v variables in view v, and let $\{\mathbf{z}^v\}$ be the collection of all correspondence vectors from the V views. We then aggregate estimates over all latent associations (some indices have been dropped to reduce clutter, if clear from context; please refer to the previous paragraph for indices):

$$\mathbb{P}\Big(\{(c,l)\} \;\Big|\; \{\{(o,x)\}\}\Big) = \sum_{\{\mathbf{z}^v\}} \mathbb{P}\Big(\{(c,l)\} \;\Big|\; \{\mathbf{z}^v\}, \{\{(o,x)\}\}\Big)\, \mathbb{P}\Big(\{\mathbf{z}^v\} \;\Big|\; \{\{(o,x)\}\}\Big) \tag{2}$$

The first term is given by the decoupled estimation problems mentioned above, and results in a closed-form posterior distribution given in the Appendix. The desired posterior distribution on the left is therefore, in exact form, a mixture over the closed-form posteriors. The problem is that the number of mixture components is exponential in M^v and V, one for each full association $\{\mathbf{z}^v\}$, so maintaining the full posterior distribution is intractable. Finding tractable approximations to this light-configuration posterior distribution is the subject of Sections 4–6.

4 A Tracking-Based Approach

If we consider the lights to be stationary targets and the views to be a temporal sequence, a target-tracking approach can be used. Tracking simultaneously solves the data association (measurement correspondence) and target parameter estimation (light colors and locations) problems. As discussed in Section 2, a wide variety of tracking algorithms exist, and in particular multiple hypothesis tracking (MHT) (Reid, 1979) has already been adopted by Elfring et al. (2013) on the problem of semantic world modeling. We provide a gist of the MHT approach and discuss a problematic issue below; readers are referred to Elfring et al. (2013) for details.

The MHT algorithm maintains, at every timestep (view) v, a distribution over all possible associations of measurements to targets up to v. At each view, MHT therefore needs to propagate *each* previous hypothesis forward with *each* possible association in view v. One way to consider this is as a tree, where nodes of depth v are associations up to view v, and a distribution is maintained on the leaves. Each view introduces a new layer of nodes, where the branching factor is the number of valid associations in that view. Without loss of generality, assume that the views are in chronological order. The distribution over associations up to view v is:

$$\begin{aligned}\mathbb{P}\Big(\{\mathbf{z}\}^{\leq v} \;\Big|\; \{\{(o,x)\}\}^{\leq v}\Big) &= \mathbb{P}\Big(\mathbf{z}^v \;\Big|\; \{\mathbf{z}\}^{<v}, \{\{(o,x)\}\}^{\leq v}\Big)\, \mathbb{P}\Big(\{\mathbf{z}\}^{<v} \;\Big|\; \{\{(o,x)\}\}^{<v}\Big) \\ &\propto \mathbb{P}\Big(\{(o^v,x^v)\} \;\Big|\; \mathbf{z}^v, \{\mathbf{z}\}^{<v}, \{\{(o,x)\}\}^{<v}\Big)\, \mathbb{P}\Big(\mathbf{z}^v \;\Big|\; \{\mathbf{z}\}^{<v}, \{\{(o,x)\}\}^{<v}\Big)\, \mathbb{P}\Big(\{\mathbf{z}\}^{<v} \;\Big|\; \{\{(o,x)\}\}^{<v}\Big)\end{aligned} \tag{3}$$

where superscript "v" indicates variables at view v only, "$\leq v$" for everything up to view v, and "$< v$" for everything up to the previous view (excluding v). The first term is the likelihood of the current view's observations, the second is the prior on the current view's correspondences given previously identified targets, and the final term is the filter's distribution from the previous views.

The likelihood term for view v follows mostly from the derivation in the Appendix. The observations are independent given the view's correspondence vector $\mathbf{z}^v$, and the likelihood is a product of M^v of the following terms:

$$\mathbb{P}\Big(o_m^v, x_m^v \;\Big|\; z_m^v = k, \{\mathbf{z}\}^{<v}, \{\{(o,x)\}\}^{<v}\Big) = \begin{cases} \frac{\phi_o^0}{b^v - a^v}, & k = 0 \\ \mathbb{P}\Big(o_m^v \;\Big|\; \{\{o\}\}_{z=k}^{<v}\Big)\, \mathbb{P}\Big(x_m^v \;\Big|\; \{\{x\}\}_{z=k}^{<v}\Big), & k \neq 0 \end{cases} \tag{4}$$

where the "$z = k$" subscript refers to observations (from previous time steps in this case) that have been assigned to the same light, as indicated by the correspondence vectors $\{\mathbf{z}\}^{<v}$. Observations corresponding to other lights are ignored because lights are assumed to be independent. The two probability terms can be found from the posterior predictive distribution (Equations 25, 29 respectively). For new targets (where k does not index an existing target), the conditioning set of previous observations will be empty, but can be likewise handled by the predictive distributions. The false positive probability ($k = 0$) follows from the observation model (Equation 1).

The prior on the current view's correspondences, the second term in Equation 3, is due to Reid (1979). Assume we know which of the existing targets are within the current field of view based on the hypothesis on previous views (this can be found by gating). Denote the indices of these targets as the size-K^v set $\{k\}^v$. Another plausible assumption used in the tracking literature, due to sensor characteristics, is that in a single view, each target can generate at most one non-spurious measurement. We will refer to this as the *one-measurement-per-object (OMPO) assumption.*

We now define validity of correspondence vectors $\mathbf{z}^v$. Recall that in this length-M^v vector, the m'th entry z_m^v is the (positive integer) target index to which (o_m^v, x_m^v) correspond, or 0 for a false positive. First, an entry in $\mathbf{z}^v$ must either be 0, a target index in $\{k\}^v$, or a new (non-existing) index; otherwise, it corresponds to an out-of-range target. Second, by the OMPO assumption, no entry may be repeated in $\mathbf{z}^v$, apart from 0 for false positives. A correspondence $\mathbf{z}^v$ is valid if and only if it satisfies both conditions.

The following quantities can be found directly from $\mathbf{z}^v$:

$$\begin{aligned} n_0 &\triangleq \text{Number of false positives (0 entries)} \\ n_\infty &\triangleq \text{Number of new targets (non-existing indices)} \\ \delta_k &\triangleq \mathbb{I}\big\{\text{Target } k \text{ is detected } (\exists m.\, z_m^v = k)\big\}, k \in \{k\}^v \\ n_1 &\triangleq \text{Number of matched targets} = M^v - n_0 - n_\infty = \sum_k \delta_k \text{ (by OMPO)} \end{aligned} \tag{5}$$

where $\mathbb{I}\{\cdot\}$ is the indicator function. Then we can split $\mathbb{P}\big(\mathbf{z}^v \,\big|\, \{\mathbf{z}\}^{<v}, \{\{(o,x)\}\}^{<v}\big)$ by conditioning on the above quantities, which are deterministic functions of $\mathbf{z}^v$:[2]

$$\mathbb{P}\big(\mathbf{z}^v\big) = \mathbb{P}\big(\mathbf{z}^v, n_0, n_\infty, n_1, \{\delta_k\}\big) = \mathbb{P}\big(\mathbf{z}^v \,\big|\, n_0, n_\infty, n_1, \{\delta_k\}\big)\, \mathbb{P}\big(n_0, n_\infty, n_1, \{\delta_k\}\big) \tag{6}$$

By the assumed model characteristics, the second term is:

$$\mathbb{P}\big(n_0, n_\infty, n_1, \{\delta_k\}\big) = \text{Binomial}\big(n_0\,;\, M^v, p_{\text{FP}}\big)\, \mathbb{P}\big(n_\infty\,;\, M^v\big)\, \mathbb{P}\big(\{\delta_k\}\big) \tag{7}$$

$$\mathbb{P}\big(\{\delta_k\}\big) = \prod_{k \in \{k\}^v} \big[p_{\text{D}}(k)\big]^{\delta_k} \big[1 - p_{\text{D}}(k)\big]^{1-\delta_k} \tag{8}$$

where $p_{\text{D}}(k)$ is the (target-specific) detection probability defined in Equation 26 in the Appendix. The number of new targets n_∞ is typically Poisson-distributed.

Determining the correspondence given the quantities above involves assigning $\mathbf{z}_m^v$ indices to the three groups of entries (of sizes n_0, n_∞, and n_1) and matching a size-n_1 subset of $\{k\}^v$ (as indicated by $\{\delta_k\}$) to the indices in the final group. A common assumption is that all assignments and matches of indices are equally likely, so the first term in Equation 6 is the reciprocal of the number of valid correspondence vectors (given n_0, n_∞, n_1, and $\{\delta_k\}$), given by:

$$n_{\text{valid}}\big(n_0, n_\infty, n_1, \{\delta_k\}\big) = \binom{M^v}{n_0, n_\infty, n_1} n_1! = \frac{M^v!}{n_0! n_\infty!} \tag{9}$$

Combining Equations 4–9 gives the necessary expressions used in the MHT filter (Equation 3).

[2]The probabilities implicitly depend on previous correspondences $\{\mathbf{z}\}^{<v}$ and observations $\{\{(o,x)\}\}^{<v}$, as shown in the second term of Equation 3, via the targets in view $\{k\}^v$ and their detection probabilities in Equation 8.

The expression for n_{valid}, which is related to the branching factor in the tree of associations that the MHT considers, highlights the complexity of this approach. To obtain the total number of valid associations, we need to also consider all possible settings of n_0, n_∞, n_1, and $\{\delta_k\}$:

$$n_{\text{total}} = \sum_{n_0=0}^{M^v} \sum_{n_\infty=0}^{(M^v - n_0)} \binom{K^v}{n_1} n_{\text{valid}}\big(n_0, n_\infty, n_1, \{\delta_k\}\big) \tag{10}$$

Even with 4 measurements and 3 within-range targets, the branching factor is 304, so considering all hypotheses over many views is clearly intractable. Many hypothesis-pruning strategies have been devised (e.g., Kurien (1990); Cox and Hingorani (1996)), the simplest of which include keeping the best hypotheses or hypotheses with probability above a certain threshold. More complex strategies to combine similar tracks and reduce the branching factor have also been considered. In the experiments of Section 7 we simply keep hypotheses with probability above a threshold of 0.01. As we will demonstrate in the experiments, an MHT filter using this aggressive pruning strategy can potentially cause irreversible association errors and make incorrect conclusions.

5 A Clustering-Based Approach

If we consider all the measurements together and disregard their temporal relationship (static world assumption), we expect the measurements to form clusters in the product space of colors and locations ($[T] \times \mathbb{R}$), allowing us to derive estimates of the number of lights and their parameters. In probabilistic terms, the measurements are generated by a mixture model, where each mixture component is parameterized by the unknown parameters of a light. Since the number of lights in the world is unknown, we also do not want to limit the number of mixture components a priori.

As mentioned in Section 2, the Dirichlet process mixture model (DPMM) supports an unbounded number of mixture components. The Dirichlet process (DP) acts as a prior on *distributions* over the cluster parameter space. Teh (2010) provides a good review of DPs and its application to mixture models. From a generative perspective, a random distribution G over cluster parameters is first drawn from the DP; G is discrete with probability one (but possibly with unbounded support). For each measurement, a (possibly-repeated) set of cluster parameters is drawn from G, and data is then drawn according to the corresponding observation model given by the parameters. Although the model can potentially be infinite, the number of clusters is finite in practice, as they will be bounded by the total number of measurements (typically significantly fewer if the data exhibits clustering behavior). The flexibility of the DPMM clustering model lies in its ability to 'discover' the appropriate number of clusters from the data.

We now derive the DPMM model specifics and inference procedure for the colored-lights domain. A few more assumptions need to be made and parameters defined. Our model assumes that the cluster parameter distribution G is drawn from a DP prior $\text{DP}(\alpha, H)$, where H is the base distribution and α is the concentration hyperparameter (controlling the similarity of G and H, and also indirectly the number of clusters). H acts as a 'template' for the DP, and is hence also a distribution over the space of cluster parameters. We set it to be the product distribution of π, the prior on colors c, and a normal-gamma distribution over the location l and its observation precision τ (see the Appendix for details on this latter distribution, as well as an interpretation of the subscripted hyperparameters):

$$H(c, l, \tau) \triangleq \pi_c \,\text{NormalGamma}\,(l, \tau\,;\, \lambda_0, \nu_0, \alpha_0, \beta_0) \tag{11}$$

To accommodate false positives, which occur with probability p_{FP}, we scale G from the DP prior by a factor of $(1 - p_{\text{FP}})$ for true positives, and let the remaining probability mass correspond to a parameter-less cluster to which all false positives are assigned.

To illustrate the analogy of the DPMM to finite mixture models, we express the DP prior in an equivalent form based on the stick-breaking construction (Sethuraman, 1994). The idea is that the sizes of clusters are determined by a random process that first selects some proportion β_1 of the unit interval ('breaks the stick'), where $\beta_1 \sim \text{Beta}\,(1, \alpha)$, and defines that to be the size of the first cluster (as a proportion). Smaller α tends to result in larger sticks earlier in the process, hence fewer clusters are preferred. This process is then recursively applied *ad infinitum* to the remaining length-$(1 - \beta_1)$ stick, resulting in a countably infinite subdivision of the interval. A corresponding infinite sequence of cluster parameters $\{(c_k, l_k, \tau_k)\}$ is drawn from the base distribution H and associated with each stick. The stick-breaking process is typically denoted by:

$$\{\beta_k\} \sim \text{Griffiths-Engen-McCloskey}(\alpha)\ ;\ \{(c_k, l_k, \tau_k)\} \sim H \tag{12}$$

By defining $G(c, l, \tau) \triangleq \sum_{k=1}^{\infty} \beta_k\ \mathbb{I}\left[(c, l, \tau) = (c_k, l_k, \tau_k)\right]$, i.e., the sum of stick weights with corresponding parameters equal to (c, l, τ), G is a distribution over the cluster parameters and is distributed as $\text{DP}(\alpha, H)$. Different draws in the stick-breaking process may lead to G having support on different numbers of cluster parameter atoms, whereas a finite mixture model can be viewed as the case where G has a fixed number of atoms in its support.

Once $\{\beta_k\}$ and $\{(c_k, l_k, \tau_k)\}$ are drawn from the DP prior, the rest of the generative process is:

$$\begin{aligned}
\theta_k &= \begin{cases} p_{\text{FP}}\ , & k = 0 \\ (1 - p_{\text{FP}})\,\beta_k\ , & k \neq 0 \end{cases} && \text{Cluster proportions (with FPs)} \\
z_m^v &\sim \theta\ ; \qquad m \in [M^v]\ ,\ v \in [V] && \text{Cluster assignment (for each obs.)} \\
o_m^v &\sim \begin{cases} \phi^0\ , & z_m^v = 0 \\ \phi^{c_z}\ , & z_m^v \neq 0 \end{cases} && \text{Color observation} \\
x_m^v &\sim \begin{cases} \text{Unif}[a^v, b^v]\ , & z_m^v = 0 \\ \mathcal{N}\left(l_z, \tau_z^{-1}\right)\ , & z_m^v \neq 0 \end{cases} && \text{Location observation}
\end{aligned} \tag{13}$$

Despite being a nice theoretical tool, an infinite collection of sticks cannot be directly handled computationally. The most straightforward way to perform inference in a DPMM is by Gibbs sampling. In particular, we derive a collapsed Gibbs sampler for the cluster correspondence variables $\{\{z_m^v\}\}$ and integrate out the other latent variables, (c, l, τ) and θ. In Gibbs sampling, we iteratively sample from the conditional distribution of each z_m^v, given all other correspondence variables (which we will denote by $\{\{z\}\}^{-vm}$). By Bayes' rule:

$$\begin{aligned}
&\mathbb{P}\Big(z_m^v = k \,\Big|\, \{\{z\}\}^{-vm}, \{\{(o, x)\}\}\Big) \\
&\quad \propto \mathbb{P}\Big(o_m^v, x_m^v \,\Big|\, z_m^v = k, \{\{z\}\}^{-vm}, \{\{(o, x)\}\}^{-vm}\Big)\ \mathbb{P}\Big(z_m^v = k \,\Big|\, \{\{z\}\}^{-vm}, \{\{(o, x)\}\}^{-vm}\Big) \\
&\quad \propto \mathbb{P}\Big(o_m^v, x_m^v \,\Big|\, \{\{(o, x)\}\}_{z=k}^{-vm}\Big)\ \mathbb{P}\Big(z_m^v = k \,\Big|\, \{\{z\}\}^{-vm}\Big)
\end{aligned} \tag{14}$$

In the final line, the first term can be found from the posterior predictive distributions described in the Appendix (Equations 25 and 29), in a similar fashion to that in the MHT (Equation 4).

Input: Observations $\{\{(o_m^v, x_m^v)\}_{m=1}^{M^v}\}_{v=1}^{V}$
Number of samples S
Output: Samples of cluster assignments $\{\{\{z_m^{v\,(s)}\}_{m=1}^{M^v}\}_{v=1}^{V}\}_{s=1}^{S}$

Init. $K := 0$; $z_m^{v\,(0)} := 0$ for all $m \in [M^v], v \in [V]$
for $s := 1$ **to** S; $v := 1$ **to** V; $m := 1$ **to** M^v **do**
  Find cluster predictive distributions and sizes using most-recent samples $\{\{z_m^{v\,(s-1)}\}\}^{-vm}$
  Compute sampling distribution (Equation 14) by multiplying Equation 4 and Equation 15 for each $k \in \{0\} \cup [K+1]$, then normalizing
  Sample $z_m^{v\,(s)}$ from sampling distribution
  if $z_m^{v\,(s)} = K+1$ **then**
    $K := K+1$
  Remove cluster with index $z_m^{v\,(s-1)}$ if it has no other assigned observations

(a) Collapsed Gibbs sampling for DPMM

Input: Observations $\{\{(o_m^v, x_m^v)\}_{m=1}^{M^v}\}_{v=1}^{V}$
Cluster penalty parameter λ
Output: Cluster assignments $\{\{\{z_m^v\}_{m=1}^{M^v}\}_{v=1}^{V}\}$

Init. $K := 1$; $z_m^v := 1$ for all $m \in [M^v], v \in [V]$
repeat
  for $v := 1$ **to** V; $m := 1$ **to** M^v **do**
    $d_m^v(k) := -\log \mathbb{P}\left(o_m^v, x_m^v \,|\, \{\{(o,x)\}\}_{z=k}\right)$ for each $k \in [K]$ (using Equation 4)
    if $\min_k d_m^v(k) > \lambda$ **then**
      $z_m^v := K+1$; $K := K+1$
    else
      $z_m^v := \arg\min_k d_m^v(k)$
until convergence
Sort clusters by size $|\{z_m^v = k\}|$, remove smallest clusters containing a total of a p_{FP}-proportion of all observations, and set associated $z_m^v = 0$

(b) Hard-clustering algorithm for DPMM, inspired by DP-means (Kulis and Jordan, 2012)

Figure 2: Two algorithms for performing inference in DPMMs, one by sampling, the other by hard clustering.

This allows us to collapse the latent cluster parameters (c, l, τ). Note that the observations being conditioned on *exclude* (o_m^v, x_m^v) and depend on the current correspondence variable samples (to determine which observations belong to cluster k).

The second term, the distribution of z_m^v given all other cluster assignments, is given by the Chinese restaurant process (CRP), which is obtained by integrating out the DP prior on θ. Together with our prior on false positives:

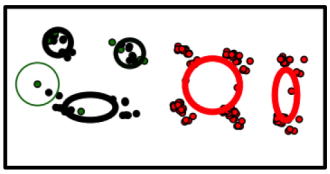
$$\mathbb{P}\Big(z_m^v = k \,\Big|\, \{\{z\}\}^{-vm}\Big) = \begin{cases} (1-p_{\text{FP}})\,\frac{N_k^{-vm}}{\alpha+N^{-vm}}\,, & k \in [K] \text{ (k exists)} \\ (1-p_{\text{FP}})\,\frac{\alpha}{\alpha+N^{-vm}}\,, & k = K+1 \text{ (k new)} \\ p_{\text{FP}}\,, & k = 0 \end{cases} \tag{15}$$

where N_k^{-vm} is the number of observations currently assigned to cluster k (excluding (o_m^v, x_m^v)), $N^{-vm} = \sum_{k=1}^{K} N_k^{-vm}$ is the total number of non-false-positive observations across all views, and K is the number of instantiated clusters. This expression illustrates the role of the concentration parameter α from a sampling perspective: larger α leads to more frequent creation of new clusters.

By combining Equations 14 and 15, we can sample from the conditional distribution of individual correspondences z_m^v. Although the model supports an infinite number of clusters, the modified CRP expression (Equation 15) shows that we only need to compute $K'+2$ values for one sampling step, where K' is the number of existing clusters with $N^{-vm} > 0$, which is finite, since clusters without data are removed. One sampling sweep over all correspondence variables $\{\{z_m^v\}\}$ constitutes one sample from the DPMM. Given the correspondence sample, finding the posterior configuration is simple. Each non-empty cluster corresponds to a light. For each cluster, applying Equations 24 and 27 from the Appendix to the cluster's associated data provides the posterior distributions on the light's color and location (with observation model precision) respectively. The posterior marginal distribution on the light's location is a t-distribution with parameters given in Equation 28.

Although Gibbs sampling is a conceptually simple inference method for recovering the posterior

(a) Single viewpoint



(b) DPMM sample clustering

(c) DPMM-FullView sample

Figure 3: A real-world example demonstrating issues with the DPMM approach from Section 5. **(a)** A scene containing many instances of the same object type, viewed from above. The relative proximity of similar objects make them easily confusable for data association. **(b)** The DPMM approach performs particularly poorly in this scenario because it ignores false-negative information and the one-measurement-per-object constraint. One sample from the posterior distribution is shown by the thick ellipses, centered around the mean cluster locations, with sizes depicting posterior variance and colors depicting object type (red = red soup can, black = orange baking-soda box); the small dots show raw measurements. Ignoring FNs causes spurious clusters (e.g., left-most thin green circle) to be more likely, since they are not discounted by absences in other views. Ignoring the OMPO assumption causes measurements from similar nearby objects to be aggregated into a single cluster, even if they were observed together in a single view, as was the case for the four soup cans in the middle. **(c)** By taking into account view-level information and constraints, the DPMM-FullView method described in Section 6.1 recovers the correct interpretation of the scene.

distribution in the DPMM, it is relatively inefficient because it requires a substantial number of samples to reach convergence. Kulis and Jordan (2012) recently proposed an elegant hard-clustering method, DP-means, that produces a single clustering assignment. The algorithm is derived from analyzing the small-variance asymptotics of the DPMM Gibbs sampler, and bears great resemblance to k-means. Like k-means, data points are assigned to their closest cluster centers, with the exception that points farther than λ away from all existing clusters are instead assigned to instantiate a new cluster. The process is repeated until convergence, which is guaranteed. The original presentation involved only Gaussian-distributed cluster means. Figure 2(b) shows our extension to the algorithm, which handles the discrete color parameter and also false positives. Although this method produces a single assignment instead of a distribution, we will use it in the next section to initialize algorithms that handle more data association constraints.

6 Incorporating View-Level Information and Constraints

The DPMM-based solution to the colored-lights problem is a straightforward application of the DPMM, but ignores two fundamental pieces of information:

- **Visible region information and false negatives (FN)**: The DPMM does not consider the field of view $[a^v, b^v]$, and hence neither which clusters are visible when a measurement is made. Clusters that are frequently visible but only sporadically detected suggest that there may in fact be no light there, that the detections were errors. Because the DPMM does not consider this, it may posit a cluster for a spurious measurement when its absence in other views would have suggested otherwise. It may also assign a measurement to a cluster that should not be visible from the current view, although this case is less likely to occur.

- **One-measurement-per-object (OMPO) assumption**: When two lights of the same color are placed close to each other, they are easily confusable. The only way to distinguish between them is if both are consistently detected together. Then, by the OMPO assumption, the two detections cannot be assigned to the same light, so the second detection must come from a

second light, or be an FP. With sufficient views, two clusters emerge. Because the DPMM ignores the OMPO assumption, it may associate both detections to the same cluster. In fact, the DPMM generally prefers larger clusters (instead of two small ones) due to the 'rich gets richer' phenomenon in the Chinese restaurant process (Equation 15).

Figure 3 illustrates a real-world example of both issues.

The above issues are consequences of the DPMM's conditional independence assumptions. To see this, consider the concrete example depicted in Figure 4, where we wish to sample cluster assignments for an entire view's $M^v = 4$ measurements. The DPMM Gibbs sampler samples the cluster assignment for each measurement *individually*, as shown in Figure 4(b). This causes the two right-most measurements to be assigned to the same cluster, a violation of the OMPO assumption. The assumption states that *at most one* measurement in a single view can be assigned to each cluster; this view-level constraint cannot be incorporated on the level of individual measurements. Likewise, a false negative only arises if *none* of the measurements in a view are assigned to a cluster within the field of view. To handle these constraints we must couple the measurements and sample their assignments *jointly*.

6.1 DPMM-FullView

More formally, instead of sampling a view v's correspondence variables $\{z_m^v\}_{m=1}^{M^v}$ one by one, we consider sampling from the conditional distribution of the *joint* correspondence vector $\mathbf{z}^v$:

$$\mathbb{P}\Big(\mathbf{z}^v \;\Big|\; \{\mathbf{z}\}^{-v}, \{\{(o,x)\}\}\Big) \propto \mathbb{P}\Big(\{(o^v,x^v)\} \;\Big|\; \mathbf{z}^v, \{\mathbf{z}\}^{-v}, \{\{(o,x)\}\}^{-v}\Big)\, \mathbb{P}\Big(\mathbf{z}^v \;\Big|\; \{\mathbf{z}\}^{-v}\Big) \tag{16}$$

Like the previous two sections, the first term is an observation likelihood term that factors into a product of M^v terms, each of which is similar to Equation 4. The second term is the DP conditional distribution on $\mathbf{z}^v$, and can be found by repeated application of the CRP (Equation 15):

$$\mathbb{P}_{\text{DPMM}}\Big(\mathbf{z}^v \;\Big|\; \{\mathbf{z}\}^{-v}\Big) = \mathbb{P}\Big(z_{M^v}^v \;\Big|\; z_{M^v-1}^v, \ldots, z_1^v, \{\mathbf{z}\}^{-v}\Big)\; \ldots\; \mathbb{P}\Big(z_2^v \;\Big|\; z_1^v, \{\mathbf{z}\}^{-v}\Big)\, \mathbb{P}\Big(z_1^v \;\Big|\; \{\mathbf{z}\}^{-v}\Big) \tag{17}$$

$$= \frac{p_{\text{FP}}^{n_0}\; (1-p_{\text{FP}})^{(n_1+n_\infty)}\; \alpha^{n_\infty}\; \Big[\prod_{\{m\}_1} N_{z_m^v}^{-v}\Big]}{\prod_{m'=0}^{(n_1+n_\infty)-1} \alpha + N^{-v} + m'} \tag{18}$$

where n_0, n_∞, n_1 are the previously-defined functions of $\mathbf{z}^v$ (Equation 5), and $\{m\}_1$ is the set of indices that are matched to existing targets (i.e., $n_1 = |\,\{m\}_1\,|$).

To see the derivation, consider the known values of n_0, n_∞, n_1 given $\mathbf{z}^v$. This means that there must be n_0 entries in $\mathbf{z}^v$ with value 0, n_∞ entries with a new positive value, and n_1 entries with an existing positive value. These three types of entries correspond exactly to the cases for the CRP, hence in Equation 17, n_0 of the terms must be p_{FP}, and so on. N^{-v} is the total number of non-view-v, non-FP observations, and $N_{z_m^v}^{-v}$ is the number of observations assigned to the cluster with index equal to the value of z_m^v, excluding view v. The latter type of counts are used in the CRP case when the assignment z_m^v corresponds to an existing cluster index k. In general, $N_k^{-vm} \geq N_k^{-v}$ (former from CRP, latter discussed above), so the expression in Equation 18 does not hold in general. However, because of the OMPO assumption, no other observation in view v could be assigned to cluster k, so in fact $N_k^{-vm} = N_k^{-v}$, and Equation 18 holds in our case.

Equation 16 is essentially the product of M^v conditional distributions used in the DPMM, and does not yet achieve our goal of incorporating FNs and the OMPO assumption. To use FNs

and field-of-view information, we take inspiration from the MHT formulation, and first suppose we knew which K^v of the existing K lights are within the field of view, i.e., $\{k\}^v$ from Section 4. This, together with $\mathbf{z}^v$, allows us to determine the detection indicator variables $\{\delta_k\}$ (Equation 5) and their probabilities (Equation 8). For the OMPO constraint, we simply assign zero probability to violating correspondences. We combine the additional information with the DPMM-based conditional distribution above (Equation 18) in a conceptually simple fashion:

$$\mathbb{P}_{\text{FullView}}\Big(\mathbf{z}^v \,\Big|\, \mathbf{z}^{-v}, \{k\}^v\Big) \propto \mathbb{P}_{\text{DPMM}}\Big(\mathbf{z}^v \,\Big|\, \mathbf{z}^{-v}\Big)\, \mathbb{P}\Big(\,\{\delta_k\}\,\Big)\, \mathbb{I}\Big[\mathbf{z}^v \text{ satisfies OMPO}\Big] \tag{19}$$

The final term evaluates to 1 if the joint correspondence satisfies the OMPO assumption, and 0 otherwise. Hence by construction the correspondence variables sampled from this conditional distribution will incorporate the FN information and OMPO constraint.

To use $\mathbb{P}_{\text{FullView}}$ as the prior over $\mathbf{z}^v$ in Equation 16, we must remove the assumption that we know $\{k\}^v$. The correct Bayesian approach is to integrate over the posterior distribution of the lights' locations, which are independent t-distributions, given by Equation 28 in the Appendix. Although this is intractable, it can be approximated by sampling the lights' locations, which is simple for t-distributions, then averaging the subsequent probabilities from Equation 19. In practice, we found that using the posterior mean location was sufficient, i.e., including light k if $\nu_k \in [a^v, b^v]$.

Although $\mathbb{P}_{\text{FullView}}$ combines all the desired information, the inherent difficulty is hidden in the '$\propto$' sign. The distribution first needs to be normalized before we can sample from it, which is inefficient now because the support of the distribution is the set of correspondence vectors satisfying the OMPO assumption. The OMPO constraint fully couples the measurements' cluster assignments, and all assignments must be considered jointly, as depicted in Figure 4(d). We have essentially reverted to the high branching factor of the MHT! In the Figure 4 example, $\mathbb{P}_{\text{FullView}}$ must be evaluated for 304 different values of $\mathbf{z}^v$, compared to the $4 \times 5 = 20$ required for the DPMM.

6.2 DPMM-Factored

A closer look at the nature of the OMPO violation suggests a potential approximation to $\mathbb{P}_{\text{FullView}}$. In Figure 4(c), the violation is caused by *only* the two right-most measurements; the two measurements on the left are not easily confusable with the others and hence are easy to handle from a data association perspective. This suggests coupling *only* those measurements that cause OMPO violations, and assume that violations involving other measurements are unlikely. Taking this a step further, we can even consider the other measurements *independently*, as in the DPMM, essentially splitting the view into three independently-considered components, as depicted in Figure 4(e).

More generally, suppose we can partition each view's set of measurements into 'violating' subsets, where all OMPO violations are contained within a single subset, with high probability. That is, a good partition has the property that any two measurements belonging to different subsets will have low probability of being assigned to the same cluster (and hence causing an OMPO violation). Let $\mathcal{P}^v$ denote such a partition on the measurement indices in view v, and let $\{\mathbf{z}^v|_p\}_{p\in\mathcal{P}^v}$ denote the restrictions of $\mathbf{z}^v$ to each subset $p \in \mathcal{P}^v$ (i.e., $\mathbf{z}^v|_p$ represents the collection of correspondence variables $\{z^v_m\}_{m\in p}$). Then we can approximately split the OMPO constraint over the partition:

$$\mathbb{I}\Big[\mathbf{z}^v \text{ satisfies OMPO}\Big] \approx \prod_{p\in\mathcal{P}^v} \mathbb{I}\Big[\mathbf{z}^v|_p \text{ satisfies OMPO}\Big] \tag{20}$$

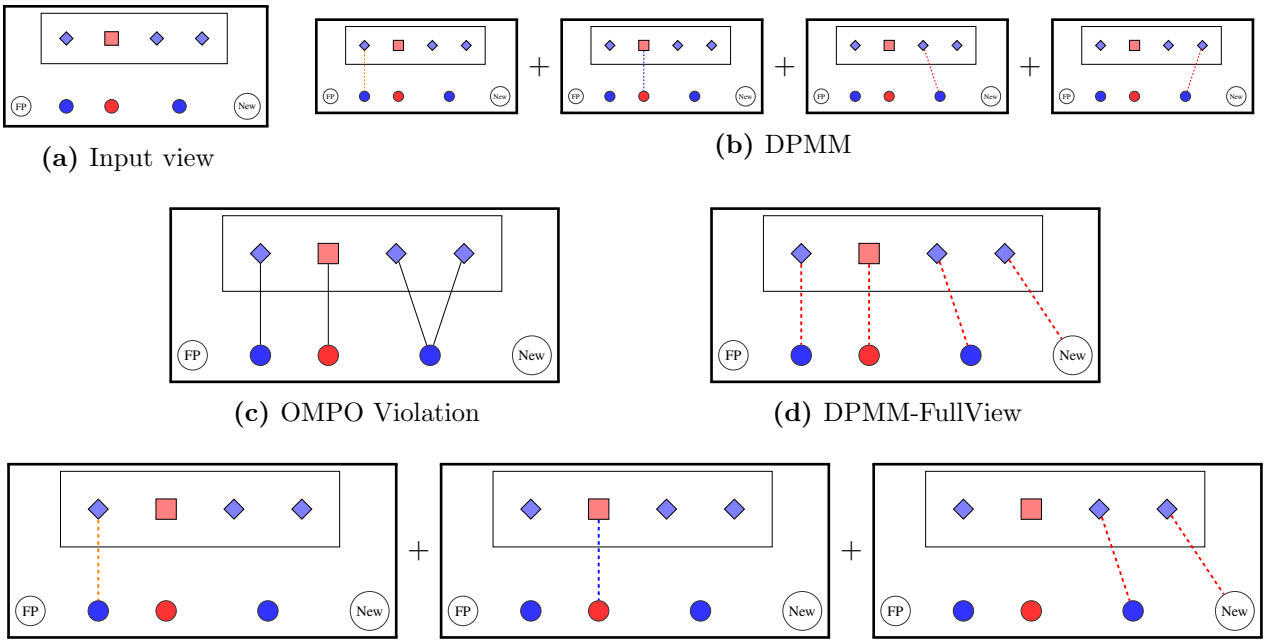


Figure 4: A concrete example for illustrating concepts in Section 6. **(a)** Each thick outer box depicts measurements in the same single view (inner box), and the clusters that each measurement can be assigned to (row below inner box). The view we consider has 4 measurements of lights' locations and colors. The 3 existing clusters within the field of view are shown as colored circles (these were determined from other views). Measurements can also be assigned to the two 'clusters' to the left and right, for false positives and new clusters respectively. The task is to assign one of the 5 clusters in the bottom row to each measurement in the inner box. **(b)** The DPMM samples cluster assignments for each measurement independently. **(c)** This causes potential violations of the one-measurement-per-object (OMPO) assumption, where each cluster generates at most one observation within each view. **(d)** One solution is to consider all measurement assignments in the view jointly. However, as explained in Section 6.1, this is inefficient. **(e)** A more efficient approximation is derived in Section 6.2 by jointly considering *only* measurements that are OMPO-violating. Measurements that are unlikely to cause constraint violation, such as the two left ones in the example, are considered independently. This provides a trade-off between DPMM and DPMM-FullView.

Returning to Figure 4(c), the most-refined partition contains three subsets, where the sole non-singleton contains the two right-most OMPO-violating measurements.

To make inference more efficient, we use the partition $\mathcal{P}^v$ to split $\mathbb{P}_{\text{FullView}}$. Ultimately, we want to express the right-hand side of Equation 19 as a *product of independent factors*, each corresponding to one subset of measurements in the partition. Because the terms are independent, sampling over the conditional distribution can be performed by sampling each factor and combining the results. Each factor is normalized over its significantly-smaller set of valid correspondence vectors, thereby improving on the bottleneck step in Gibbs sampling for DPMM-FullView.

We now consider how to factor the other two terms in $\mathbb{P}_{\text{FullView}}$ (Equation 19). $\mathbb{P}_{\text{DPMM}}\left(\mathbf{z}^v \,|\, \mathbf{z}^{-v}\right)$ is already a product over the conditional distributions of the correspondence variables, which is clear from Equation 17. By collecting terms according to the partition $\mathcal{P}^v$, we can write:

$$\mathbb{P}_{\text{DPMM}}\left(\mathbf{z}^v \,\middle|\, \mathbf{z}^{-v}\right) = \prod_{p \in \mathcal{P}^v} \mathbb{P}_{\text{DPMM}}\left(\mathbf{z}^v|_p \,\middle|\, \mathbf{z}^{-vp}\right) \tag{21}$$

The remaining term, $\mathbb{P}\left(\{\delta_k\}\right)$, is also a product of distributions, but over the set of lights $\{k\}^v$ that

Input: Observations $\{\{(o_m^v, x_m^v)\}_{m=1}^{M^v}\}_{v=1}^{V}$
Fields of view $\{[a^v, b^v]\}$
Number of samples S
Output: Samples of cluster assignments $\{\{\{z_m^{v\,(s)}\}_{m=1}^{M^v}\}_{v=1}^{V}\}_{s=1}^{S}$

1: Init. $K := 0$; $z_m^{v\,(0)} := 0$ for all $m \in [M^v], v \in [V]$
2: **for** $s := 1$ **to** S; $v := 1$ **to** V **do**
3: Find cluster post./pred. distributions and sizes using most-recent samples $\{\mathbf{z}^{(s-1)}\}^{-v}$
4: Find $\{k\}^v$, the lights within field of view: include light k iff. mean location $\nu_k \in [a^v, b^v]$
5: **for each** valid correspondence vector $\mathbf{z}^v$ (from total given by Equation 10) **do**
6: Compute sampling 'probability' $\mathbb{P}\left(\mathbf{z}^v \,\middle|\, \{\mathbf{z}^{(s-1)}\}^{-v}, \{\{(o,x)\}\}, \{k\}^v\right)$ (unnormalized; using Equations 8, 16–19)
7: Sample $\mathbf{z}^{v\,(s)}$ from normalized distribution
8: $K := K + n_\infty(\mathbf{z}^{v\,(s)})$
9: Remove clusters with no observations

(a) Collapsed Gibbs sampling for DPMM-FullView

Input: Observations, fields of view, num. samples
Cluster penalty parameter λ
Output: Samples of cluster assignments

1: Init. K, $\{\{z_m^{v\,(0)}\}\}$ from DP-means (Figure 2(b))
2: **for** $v := 1$ **to** V **do**
3: $\mathcal{P}^v :=$ Partition induced by transitive closure of R^v, where $(i,j) \in R^v$ iff. $z_i^{v\,(0)} = z_j^{v\,(0)} \neq 0$
4: **for** $s := 1$ **to** S; $v := 1$ **to** V **do**
5: Find cluster post./pred. distributions and sizes using most-recent samples $\{\mathbf{z}^{(s-1)}\}^{-v}$
6: **for each** subset of indices $p \in \mathcal{P}^v$ **do**
7: Find assigned lights $\{k\}^v|_p$: $k \in \{k\}^v|_p$ iff. $\min_{i\in p} \|\nu_k - x_i^v\| < \min_{j\notin p} \|\nu_k - x_j^v\|$
8: Sample $\mathbf{z}^{v\,(s)}|_p$ by performing steps 5–7 of DPMM-FullView (Figure 5(a)), using $\{k\}^v|_p$ and $\{(o_m^v, x_m^v)\}_{m\in p}$
9: $\mathbf{z}^{v\,(s)} :=$ Concatenation of $\left\{\mathbf{z}^{v\,(s)}|_p\right\}_{p\in\mathcal{P}^v}$
10: Update clusters (DPMM-FullView steps 8–9)
11: Agglomerate elements in partitions with OMPO violations (steps 2–3)

(b) Partitioning and sampling for DPMM-Factored

Figure 5: Two modifications to the DPMM Gibbs sampling algorithm (Figure 2(a)), by incorporating view-level information and constraints (DPMM-FullView), and using an efficient factored approximation (DPMM-Factored).

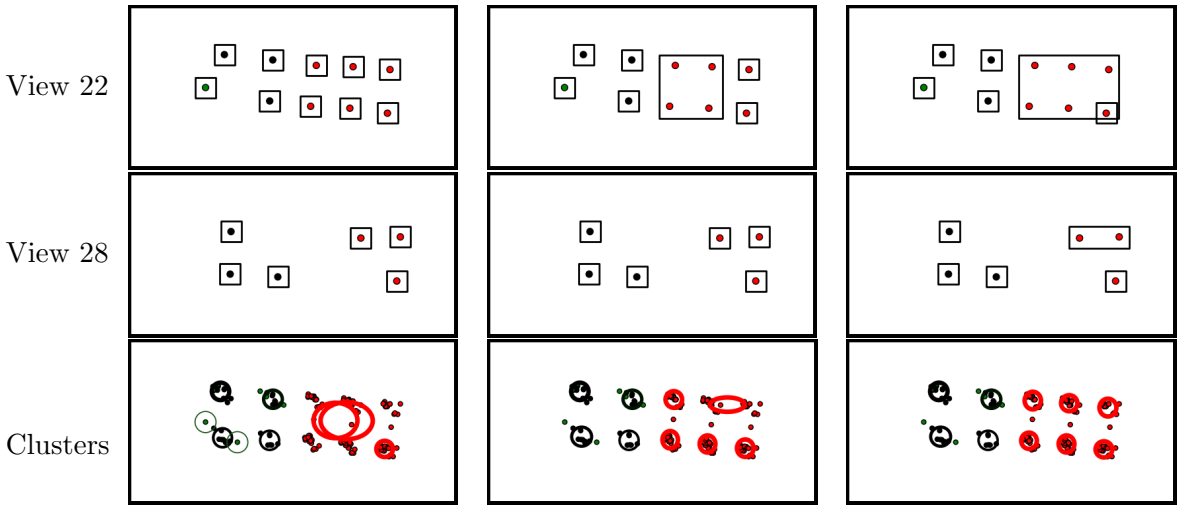
are within the field of view. Unfortunately, this cannot be immediately written as a product over the partition. We therefore make a further approximation by assigning each light to some $p \in \mathcal{P}^v$. In particular, for each light, the closest measurement (from the light's posterior mean location) was determined, and the light was assigned to the partition subset containing the measurement. Another way to view this scheme is that the partition $\mathcal{P}^v$ induces a partition over Voronoi cells in the space of location measurements (bounded by the field of view), and lights are assigned to partition elements according to the cells that their posterior mean locations are situated in.

Putting everything together, we arrive at the following *factored* approximation:

$$\mathbb{P}_{\text{Factored}}\left(\mathbf{z}^v \,\middle|\, \mathbf{z}^{-v}, \{k\}^v\right) \propto \prod_{p\in\mathcal{P}^v} \mathbb{P}_{\text{DPMM}}\left(\mathbf{z}^v|_p \,\middle|\, \mathbf{z}^{-vp}\right) \mathbb{P}\Big(\{\delta_k\}|_p\Big) \mathbb{I}\Big[\mathbf{z}^v|_p \text{ satisfies OMPO}\Big] \tag{22}$$

where $\{\delta_k\}|_p$ denotes the restriction of $\{\delta_k\}$ to the lights assigned to subset p according to the scheme described above. This form makes clear that each factor can be normalized and sampled independently. With a good partition, the large joint computation in DPMM-FullView is broken into several smaller ones within each element of $\mathcal{P}^v$. For the concrete example in Figure 4, the sampling process is depicted in Figure 4(e), where the partition is such that only the OMPO-violating measurement pair is considered jointly. This results in computing $5 + 5 + 22 = 32$ values, which is slightly greater than DPMM (20) but significantly fewer than DPMM-FullView (304).

One issues remains: Where does the partition come from? This is crucial for all factored approximations: the aggressiveness of partitioning determines the trade-off between approximation error and efficiency. On one extreme, the DPMM model is similar to a fully-factored model (but does not take into account false negatives); on the other extreme, DPMM-FullView is equivalent to



(a) Iteration 0 **(b)** Iteration 2 **(c)** Iteration 4

Figure 6: An illustration of the DPMM-Factored method being applied on the real-world example from Figure 3. The top two rows show two views (of 28) taken for the scene, with different partitions of measurements (surrounding boxes) assigned by DPMM-FullView over several iterations. The final row depicts the inferred clusters/objects and their attributes (in thick ellipses) after aggregating correspondences from the 28 views. **(a)** The correspondences are initialized by DP-means, which assumes that measurements are independent, hence partitions consist of singletons only. The resulting clusters suffer from the same issues depicted in Figure 3, as expected. In particular, the red measurements in the middle are frequently aggregated into two large clusters, which is incorrect. **(b)** In view 22, an OMPO violation is detected because the four red measurements in the middle are all previously assigned to one of the large clusters. These singleton partition elements are agglomerated to respect the assumption that OMPO violations are contained within a single subset, and are sampled together to ensure their joint correspondence does not violate the OMPO assumption. This modification significantly improves the cluster estimates, with only one error remaining. The other two red measurements on the right, and the three in view 28, are not coupled at this point because they respect the OMPO assumption so far. **(c)** The two measurements in the top right now form an OMPO violation. This shows that measurements that were previously not in violation could later become OMPO-violating, because the clusters, and therefore the probability of being assigned to them, change between iterations. The partition is updated again to couple the OMPO-violating measurements and results in the correct clustering. (The large partition element in view 22 does not contain the bottom right measurement, which is a singleton in the partition.)

a one-set partition. The example in Figure 4(c) once again provides an answer: 'violating' subsets can be found by examining clusters in the DPMM samples. Specifically, if measurements tend to be assigned to the same cluster across samples, then clearly they are strong violators and should be considered jointly. We therefore group measurements together if the proportion of samples in which they are assigned to the same cluster exceeds some threshold value. This proportion allows one to select an appropriate trade-off level.

For further efficiency in determining the partition, we also considered using the DP-means hard-clustering algorithm described at the end of Section 5. The observations were first used to quickly generate a deterministic clustering, after which the cluster assignments were examined. If two measurements within the same view were assigned to the same non-false-positive cluster, they were marked as coupled together. The partitions used by DPMM-Factored were then determined by taking the transitive closure of the coupling relation in each view. Formal details on finding this partition and determining the subsequent assigned lights can be found in Figure 5(b).

Returning to the real-world example from Figure 3, the steps taken by DPMM-Factored are

partially illustrated in Figure 6. Initially, DP-means is used to cluster the measurements, and considers correspondences independently, which is equivalent to using a partition of singletons (boxes around each measurement in iteration 0). Like the DPMM sample shown in Figure 3(b), measurements in the middle from similar nearby objects are aggregated into two large clusters. The partition is updated by examining the current correspondences in each view and grouping together measurements (in the same view) that are assigned to the same cluster, i.e., violating the OMPO assumption. This results in the large box for view 22 in iteration 2. Concretely, this means that during sampling, the correspondences for the four measurements are considered *jointly*, as a single product term in Equation 22, such that an OMPO violation will not exist *for this subset of four measurements only*. This partition expansion is considered for each view, and as a result splits the large cluster into four separate ones. However, this creates a new OMPO violation, so partition expansion is performed again, resulting in an even larger partition for iteration 4. This finally resolves all OMPO violations and identifies objects in the scene correctly.

This example also illustrates the computational advantages of using DPMM-Factored, as opposed to DPMM-FullView and MHT. Although multiple iterations are needed to converge to a partition that resolves all OMPO violations, these iterations are fast if the initial partition is aggressively fine (e.g., the all-singleton partition induced by DP-means). Our partition expansion scheme couples measurements together where necessary but no further, unlike MHT, which tends to couple more measurements even with aggressive gating, and DPMM-FullView, which couples together all measurements in a single view. For example, the four measurements on the left (black/green) tend to be sampled jointly by other approaches, but DPMM-Factored detects that they are sufficiently well-separated empirically (OMPO violations are rare) and leaves them to be sampled independently. Another example is illustrated in the final iteration for view 22, where the six measurements on the right (red) are split into a subset of five measurements, and a singleton (bottom right); other methods would consider all six together. Because the number of valid correspondence vectors to consider is combinatorial in the number of measurements (Equation 9), finer partitions directly imply more efficient inference (at the potential expense of accuracy). Using Equation 10, the total number of associations that potentially need to be considered with six measurements is $n_{\text{total}}(M^v = 6, K^v = 6) = 58576$, whereas the number for the DPMM-Factored partition is $n_{\text{total}}(M^v = 5, K^v = 5) + n_{\text{total}}(M^v = 1, K^v = 1) = 5755$, an order of magnitude less. In practice, when gating is applied for the situation shown, MHT typically evaluates 1800–2400 correspondences, whereas DPMM-Factored only considers 250-350.

7 Application to Object Type-and-Pose Estimation

As mentioned in Section 3, the colored-lights domain is representative of the semantic world-model estimation problem by considering lights as objects, and colors/locations as discrete/continuous attributes respectively. Other attributes are straightforward to incorporate as long as their predictive distributions are simple to compute. To see this, recall that in our various approaches to compute or sample from the distribution of measurement associations (Equations 3, 14, and 16), attribute measurements only appear in predictive likelihood terms (e.g., Equation 4). These predictive terms assess how well each measurement (o, x) fits with an object, whose attributes have a posterior distribution determined by all the other measurements $\{\{(o, x)\}\}_{z=k}$ currently associated with the object. Such computations are necessary for each measurement, so simplicity in calculating the predictive likelihood is crucial. However, the measured attribute values do not appear elsewhere;

in particular, the correspondence priors described in Sections 4–6 do not depend on the observed values, and can be applied as described for arbitrary numbers and types of attributes.

Object attributes such as type and pose will produce predictive distributions similar in form to that for color and location in the colored-lights domain (see the Appendix for the forms). More generally, discrete attributes will have terms resembling 'color', and continuous attributes with Gaussian-like noise will have terms resembling 'location'. If attributes are independent, we can take the product of their observation models to determine the joint posterior or predictive distributions, such as the product in Equation 4. Dependent attributes will need to be jointly considered as a single unit. For example, for pose estimates with non-diagonal error covariances, the normal-gamma prior needs to be replaced with a normal-Wishart prior. For simplicity, we assume that the error covariance is axis-aligned and use an independent normal-gamma prior for each dimension. This is partially justified by the empirical observation that our measurements do not align significantly in any particular direction (see the small dots in Figure 7, depicting individual measurements).

We applied our discussed approaches to object type-and-pose estimation on tabletop scenes, illustrated in Figure 1. We recognize that estimating object types and poses from separate object detections in the fashion proposed below is unlikely the most effective use of visual data, and that much information (e.g., image features, contextual cues, similarities between consecutive frames) are discarded in the process. However, we are ultimately interested in situations where only black-box attribute detectors are accessible. Object type-and-pose estimation was chosen as an exemplary problem because it is simple to understand and has immediate application.

Estimating object types and poses is similar to the colored-lights problem, where 'type' is a discrete attribute equivalent in form to 'color', and 'pose' is a 3-D version of 'location' with Gaussian-like observation noise. For our experiments, we placed a uniform prior on the 'type' attribute, with the $\mathbb{P}(\text{correct detection}) = 0.6$, $\mathbb{P}(\text{false negative}) = 0.1$, and the rest of the probability mass spread uniformly across other types. In the notation of Equation 1, for type i we assume that:

$$\phi_i^i = 0.6 \qquad \phi_0^i = 0.1 \qquad \phi_{j\neq i}^i = \frac{1 - \phi_i^i - \phi_0^i}{C - 1} \tag{23}$$

For 'pose', we assumed that objects are resting stably and upright above a surface, so only the (x, y, θ) positions of their reference points were considered. Further, as mentioned above, we assumed the observation noise is independent in each dimension, and placed a normal-gamma distribution on each, with the same hyperparameters as specified in the Appendix ($\alpha_0 = 10, \beta_0 = 9 \times 10^{-3}$). The observation likelihood of each measurement is similar in form to Equation 4, except with two additional terms in the product that resemble 'location' for the two extra attribute dimensions.

Detections of object type and pose came from 3-D point-cloud data obtained from a Kinect sensor mounted on a mobile robot. The object detector starts by clustering the points above a resting surface, such as a table or a shelf. It assumes that objects are separated on the surface, so that each point cloud corresponds to at most one object. For each cluster, it tries fitting point-cloud object models to the observed points, by optimizing a fitness function with respect to the object pose (x, y, θ). The z-position of the object is constrained so that the bottom of the object touches the resting surface. The optimizer tries alignments starting from several different model resting orientations based on the convex hull of the object model. The fitness function is computed as a weighted linear combination of least-squares range-image errors and point-cloud nearest-neighbor distances (approximated with a distance transform of the model point cloud). The detection pipeline is related to the system described by Glover and Popovic (2013).

For our scenarios, objects of 4 distinct types, possibly with multiple instances of the same type, were placed on a table. A robot moved around the table in a circular fashion, obtaining 20-30 views in the process; see Figures 9 and 10 for RGB images of example views (although only depth data is used to detect objects). When a view is captured, the object detection system described above is given the Kinect point cloud as input, as outputs a list of object types and (x, y, θ) poses, one for each segmented cluster in the point cloud. Figure 8 visualizes the detections (superimposed in red using the known shape models) for several views; common detection errors are also illustrated. We constructed 12 scenes of varying object and occlusion density to test our approaches; results for 5 representative scenarios are described in the next section.

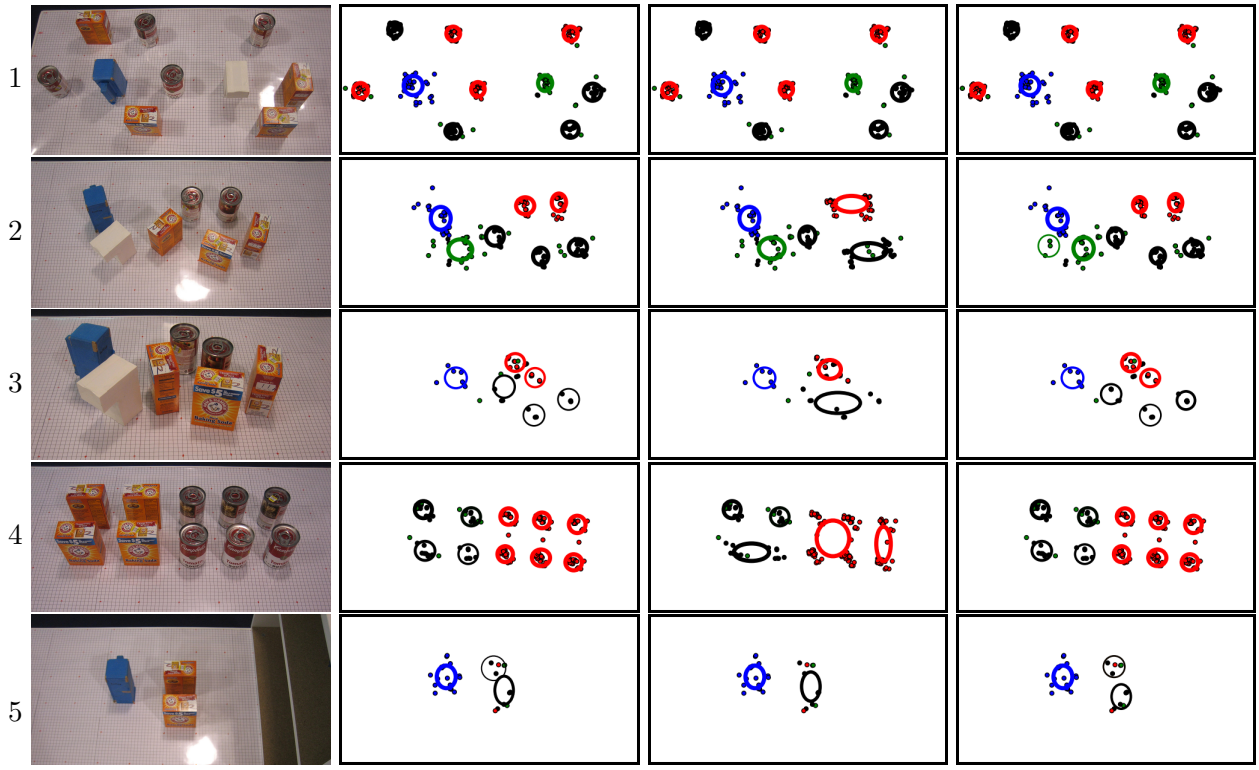
7.1 Qualitative Results

Qualitative results for 5 representative scenarios are shown in Figure 7. Images from above are for comparison convenience only; the camera's viewing height is much closer to the table height, as shown in Figures 9 and 10, so in each view only a subset of objects is detectable. We compare three approaches: multiple hypothesis tracking (**MHT** from Section 4; a re-implementation of the approach by Elfring et al. (2013)), generic DPMM clustering (**DPMM** from Section 5), and the factored approximation to DPMM-FullView (**DPMM-Factored** from Section 6.2). In Figure 7, the most likely hypothesis is shown for **MHT**, and the maximum *a posteriori* (MAP) sample (out of 100) is shown for the clustering-based approaches.

All approaches work well for scenario 1, where objects are spaced far apart. As objects of similar type are placed near each other, **DPMM** tends to combine clusters since it ignores the OMPO assumption. This is most apparent in scenario 4 (also illustrated in Figures 3 and 6), where four soup cans (red) were combined into a single large cluster. By reconsidering the OMPO assumption, **DPMM-Factored** performs significantly better and is on par qualitatively with the **MHT**, except for an extra cluster (bottom left, green) in scenario 2.

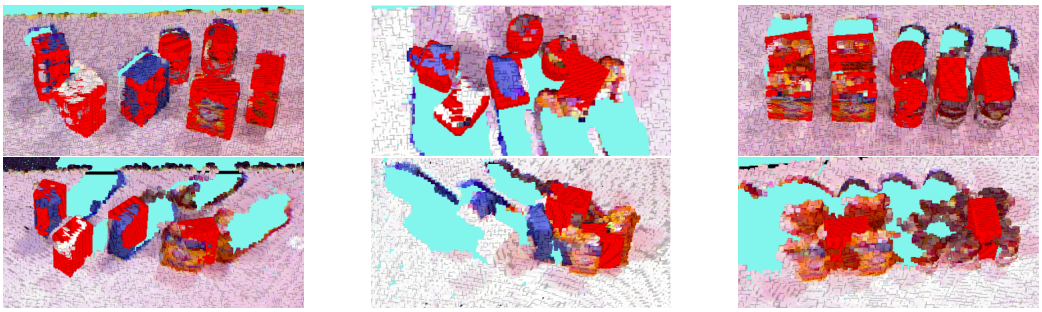
In more detail, for scenario 2, the measurements corresponding to the white L-shaped object are dispersed, causing the shown extra-cluster error to be likely. Examining more samples reveals that a significant proportion (31%) do not have the extra cluster; they just happen not to be MAP samples. This means that the estimator has significant uncertainty as to whether or not the extra object exists. Although in this case the **DPMM-Factored** MAP sample is wrong, it highlights a feature of our approach. Consider a task, e.g., grasping, that requires an accurate estimate of this object's neighborhood. Given the high uncertainty in the samples, the robot should decide to gather more observations of the region instead of operating based on the incorrect MAP sample. In contrast, the **MHT** is over 90% certain of its estimate because most other possibilities have been pruned. Although **MHT** would have been less certain as well if all hypotheses were retained during filtering, the necessary aggressive pruning tends to make **MHT** overconfident in its estimates.

Scenario 5, shown in Figure 9, highlights another difference between the tracking filter and batch approaches. There is significant occlusion early in the sequence, which throws off **MHT**, causing it to make incorrect associations which result in poor pose estimates. Here two closely-arranged boxes are placed near a shelf, such that from most views at most one of the two boxes can be seen. Only in the final views of the sequence can both be seen (final image in Figure 9). Due to the proximity of the boxes, and the fact that at most one was visible in the early views, **MHT** eventually pruned all the then-unlikely hypotheses positing that measurements came from two objects. When finally both are seen together, although a hypothesis with two orange boxes resurfaces, it is too late: the remaining association hypotheses already associate all previous measurements of the



(a) Scene from above **(b)** MHT **(c)** DPMM **(d)** DPMM-Factored

Figure 7: Qualitative results for 3 world-model estimation approaches in 5 scenarios. The bird's-eye view of the scenes is for comparison convenience only; the actual viewing height is much closer to the table. The most likely hypothesis is shown for MHT, and the maximum a posteriori sample is shown for the clustering-based approaches. Each small colored dot is a semantic (object type-and-pose) detection. Each target/cluster is depicted by an ellipse, centered at the posterior mean location. Ellipse axis lengths are proportional to the standard deviation in their respective dimensions. Ellipses are color-coded by the most likely posterior object type: red = red soup can, black = orange baking-soda box, green = white L-shaped block, blue = blue rectangular cup. Line thickness is proportional to cluster size. See text in Section 7 for qualitative comparisons.



(a) Scenario 2 **(b)** Scenario 3 **(c)** Scenario 4

Figure 8: Examples of object detections in several views (superimposed in red using the known shape model of the detected type). The top row shows cases where the detections are relatively accurate, whereas the bottom row shows cases where most detections are missing or wrong. Missing objects are typically due to occlusion by other objects. When only a small part of an object is visible, it is often not segmented properly, which then affects the fitted poses.

Figure 9: Examples of views from scenario 5. In all views except the final one, only one of the two orange baking-soda boxes was detected. This causes **MHT** to incorrectly infer that all measurements came from a single object (and prune other possibilities), whereas batch approaches such as clustering can revisit previous associations and correct them using new information, such as the final view where both boxes are finally detected.

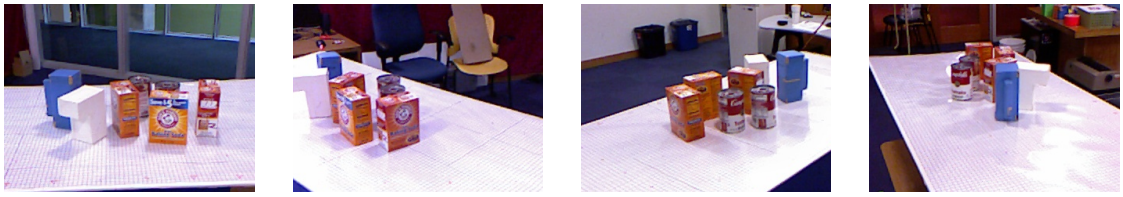


Figure 10: Examples of views from scenario 3. Objects were placed relatively close together, resulting in significant occlusion, causing the object detector to frequently miss detections or produce errors.

boxes to the same target, in turn giving an inaccurate location estimate. In contrast, **DPMM-Factored** re-examines previous associations (in the next sampling iteration) after the two boxes are seen together, and can correct such errors. One way to consider this difference is that the clustering-based methods repeatedly revisits all association decisions, whereas **MHT** prunes away most association hypotheses, and once having done so it cannot revisit a wrong decision.

7.2 Quantitative Comparisons

Quantitative metrics are given in Table 1, averaged over the association hypotheses for **MHT**, and over 100 samples for **DPMM**, **DPMM-FullView**, and **DPMM-Factored** (after discarding burn-in). We also compare against our version of the hard-clustering algorithm **DP-means**, for several different penalty parameter (λ) settings; recall that larger λ tends to lead to more, tighter clusters (see Figure 2(b), line 5 to see its role as a threshold for cluster membership). Finally, we consider a baseline approach, **Raw**, that does not perform any data association. It uses the object types and poses perceived in each view directly as a separate prediction of the objects present within the visible field of view. The metrics in the table are evaluated for each view's prediction, and the **Raw** table rows show the average value over all views.

The need for aggregating measurements across views is exemplified by **Raw**'s tendency to miss objects or confuse their types within single views. Out of the 12 scenes attempted, scenario 3, shown in Figure 10, was the most challenging for **Raw**, because objects were placed close together. This caused segmentation errors frequently occur and resulted in clear errors and unidentifiable point cloud clusters. Significant occlusion also caused missed detections. As a result, using any single view's detections is unlikely to produce an accurate estimate of objects in the world. In the scenario's sequence of 21 views, although most objects are detected (sometimes incorrectly) fewer than 5 times, the combined information is sufficient for **MHT**, **DPMM-FullView**, and **DPMM-Factored** to achieve good qualitative and quantitative results.

Table 1: Average accuracy metrics and computation wall times for the five scenarios shown in Figure 7. **Raw** is a baseline that does not perform data association; object detections are used 'as-is'. The three 'λ' rows refer to the **DP-means** algorithm, for different settings of the penalty parameter. To evaluate predicted targets and clusters against our manually-collected ground truth, for each ground truth object, the closest cluster within a 5 cm radius is considered to be the estimate of the object. If no such cluster exists, then the object is considered missed; all predicted clusters not assigned to objects at the end of the process are considered spurious. For the two parameter accuracy metrics (most-likely type, location error), parameters were only evaluated for clusters that were matched to the ground truth (i.e., for the clusters counting towards the number of correct clusters), so values are not comparable across all methods. Computation wall times were computed on a single core of an 2.3 GHz Intel Core i7 processor, using implementations in Python. Correspondences evaluated and computation times are not provided for **Raw** since no processing in the measurements is required. The '†' symbol for **MHT** and **DPMM-FullView** in scenarios 1 and 4 indicate that external guidance was necessary, in the form of manually splitting the views (each into 2-3 parts, consisting of 3-6 measurements each). Without such help, both algorithms take over an hour to complete for scenarios 1 and 4; scenario 2 is an example of a case of moderate complexity, where no guidance was provided.

Metric →	Num. correct objects (TPs)					Num. missed objects (FNs)					Num. spurious clusters (FPs)					F_1 score $= 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$				
Scenario	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
Raw	8.0	3.3	1.6	5.3	1.0	2.0	3.7	5.4	4.7	2.0	0.8	1.3	0.3	0.1	0.7	0.85	0.53	0.32	0.64	0.74
$\lambda = -1$	4.0	2.0	2.0	1.0	0.0	6.0	5.0	5.0	9.0	3.0	1.0	1.0	**0.0**	2.0	1.0	0.53	0.40	0.44	0.15	0.00
$\lambda = -2.5$	8.0	2.0	5.0	4.0	3.0	2.0	5.0	2.0	6.0	**0.0**	2.0	4.0	**0.0**	1.0	**0.0**	0.80	0.31	0.83	0.53	**1.00**
$\lambda = -4$	**10.0**	**7.0**	6.0	**6.0**	**3.0**	**0.0**	**0.0**	**1.0**	4.0	**0.0**	4.0	5.0	2.0	3.0	1.0	0.83	0.74	0.80	0.63	0.86
MHT	**10.0**	**7.0**	**6.0**	**10.0**	2.4	**0.0**	**0.0**	**1.0**	**0.0**	0.6	**0.0**	**0.0**	**0.0**	**0.0**	0.6	**1.00**	**1.00**	**0.92**	**1.00**	0.81
DPMM	8.0	2.1	2.1	4.0	2.7	2.0	4.9	4.9	6.0	0.3	1.0	2.7	**0.0**	1.0	**0.0**	0.84	0.36	0.46	0.53	0.94
FullView	**10.0**	**7.0**	**6.0**	**10.0**	**3.0**	**0.0**	**0.0**	**1.0**	**0.0**	**0.0**	0.1	1.5	0.2	**0.0**	**0.0**	**1.00**	0.91	0.91	**1.00**	**1.00**
Factored	**10.0**	**7.0**	**6.0**	**10.0**	2.9	**0.0**	**0.0**	**1.0**	**0.0**	0.1	0.8	3.1	0.3	**0.0**	**0.0**	0.96	0.82	0.90	**1.00**	0.97

Metric →	Most-likely type is correct (%)					Location estimate error (cm)					Num. correspondences evaluated ($\times 10^3$)					Computation wall time (s)				
Scenario	1	2	3	4	5	1	2	3	4	5	1†	2	3	4†	5	1†	2	3	4†	5
Raw	98	93	67	85	56	2.5	2.7	1.9	2.2	2.1	N/A					N/A				
$\lambda = -1$	100	100	50	100	-	2.0	2.5	3.8	1.1	-	4.48	2.50	0.21	1.70	0.03	24.3	7.4	1.0	8.6	0.6
$\lambda = -2.5$	100	100	80	100	100	2.2	2.4	2.4	1.0	2.2	9.09	2.65	0.76	2.16	0.25	19.4	6.2	0.5	6.7	0.6
$\lambda = -4$	100	100	100	100	100	2.1	2.6	1.7	1.3	2.4	12.7	5.58	0.86	3.80	0.21	12.8	5.3	0.1	4.0	0.1
MHT	100	100	83	100	100	2.1	2.8	1.8	1.3	2.6	5.72†	195	18.3	105†	0.25	16.9†	593	41.6	211†	0.5
DPMM	100	100	56	100	95	2.2	2.4	3.8	1.0	2.7	69.9	23.5	3.37	22.0	2.88	144	39.8	3.6	37.3	4.2
FullView	100	100	90	100	100	2.0	2.6	1.6	1.4	2.2	44.9†	582	19.3	167†	2.62	171†	1346	33.7	278†	5.3
Factored	100	100	88	100	96	2.1	2.6	1.6	1.3	2.4	8.91	6.84	1.48	19.0	0.98	127	64.9	9.9	90.6	4.5

Summarizing the results in Table 1, we find that **DPMM** overcomes noise in a single view by clustering across views, but still misses many objects because it ignores the OMPO assumption and agglomerates nearby similar objects. **DPMM-FullView** respects this constraint and performs significantly better, missing few objects while maintaining accuracy in the posterior type-and-pose estimates. **DPMM-Factored** performs similarly in quality, with an increase in spurious clusters. However, this minor hit in quality comes with an order-of-magnitude computational improvement compared to **DPMM-FullView**. The clustering approaches tend to have more spurious clusters because we chose hyperparameters that encourage positing new clusters and faster exploration of the association space, but this can be corrected at the expense of convergence speed. The **MHT** achieves the overall best quantitative performance, but in most cases is only marginally better than **DPMM-Factored**, an improvement that comes at a high computational expense, and potentially introduces filtering-related overconfidence issues mentioned earlier.

8 Discussion

We have presented several clustering-based data association approaches for estimating semantic world models. We use Dirichlet process mixture models (DPMM) as our underlying framework. However, DPMMs perform poorly in their generic form because they ignore crucial view-level information and constraints. Two improvements were therefore developed by incorporating the OMPO constraint exactly and approximately respectively. In preliminary experiments based on tabletop object type-and-pose estimation, the latter approach (**DPMM-Factored**) achieved performance comparable to a tracking-based approach (**MHT**) using a fraction of the computation time.

If only a single posterior association is needed (instead of a distribution), the hard-clustering algorithm **DP-means** performs surprisingly well, and is much faster than all the other methods. However, performance depends heavily on setting this parameter appropriately. To some extent, this could be alleviated by starting with an overly-conservative (large) value of λ, with few clusters and many OMPO violations, and then gradually decreasing λ until most violations are resolved. This will still lead to an abundance of spurious clusters, as seen in the results. Using view-level information to merge *clusters* (instead of sampling *cluster assignments*) may prove beneficial.

As discussed in the introduction, semantic world models are useful in many object-centric tasks, involving a diverse set of attributes. We are currently exploring applications involving attributes beyond object type and pose. To be truly applicable, world models must also cope with objects moving over extended periods of time. Since the presented sampling procedure for inference iterates through all views, it is at present impractical to apply it to the entirety of the robot's observation history. Instead, a hybrid approach combining the benefits of filtering and batch data association is desirable. Extending our framework to handle temporal dynamics while maintaining tractability over long horizons is the subject of future work.

Funding

This work was supported in part by the NSF under Grant No. 1117325. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. We also gratefully acknowledge support from ONR MURI grant N00014-09-1-1051, from AFOSR grant FA2386-10-1-4135, and from the Singapore Ministry of Education under a grant to the Singapore-MIT International Design Center.

Appendix: Posterior and predictive distributions for a single light

In this appendix, we verify the claim from Section 3 that finding the posterior and predictive distributions on color and location for a single light is straightforward, given that we know which observations were generated by that light. Let $\{(o, x)\}$ denote the set of light color-location detections that correspond to a light with unknown parameters (c, l). Color and location measurements are assumed to be independent given (c, l) and will be considered separately. We assume a known discrete prior distribution $\pi \in \Delta^{(C-1)}$ on colors, reflecting their relative prevalence. Using the color noise model (Equation 1), the posterior and predictive distributions on c are:

$$\mathbb{P}\left(c \,|\, \{o\}\right) \propto \mathbb{P}\left(\{o\} \,|\, c\right)\ \mathbb{P}\left(c\right) \propto \left[\prod_{i \in \{o\}} \phi_i^c\right]\ \pi_c \tag{24}$$

$$\mathbb{P}\left(o' \,\middle|\, \{o\}\right) = \sum_{c=1}^{C} \mathbb{P}\left(o' \middle| c\right)\ \mathbb{P}\left(c | \{o\}\right) = \sum_{c=1}^{C} \phi_{o'}^c\ \mathbb{P}\left(c \,|\, \{o\}\right) \tag{25}$$

We can use this to find the light's probability of detection:

$$p_{\mathrm{D}} \triangleq 1 - \mathbb{P}\left(o' = 0 \,\middle|\, \{o\}\right) = 1 - \sum_{c=1}^{C} \phi_0^c\ \mathbb{P}\left(c \,|\, \{o\}\right) \tag{26}$$

Unlike the constant false positive rate p_{FP}, the detection (and false negative) rate is dependent on the light's color posterior.

For location measurements, we emphasize that both the mean l *and* precision $\tau = \frac{1}{\sigma^2}$ of the Gaussian noise model is unknown. Modeling the variance as unknown allows us to attain a better representation of the location estimate's empirical uncertainty, and not naïvely assume that repeated measurements give a known fixed reduction in uncertainty each time. We use a standard conjugate prior, the distribution NormalGamma$(l, \tau; \lambda, \nu, \alpha, \beta)$. The typical interpretation of normal-gamma hyperparameters is that the mean is estimated from λ observations with mean ν, and the precision from 2α observations with mean ν and variance $\frac{\beta}{\alpha}$. It is well known (e.g., Bernardo and Smith (1994)) that after observing n observations with sample mean $\hat{\mu}$ and sample variance $\hat{s}^2$, the posterior is a normal-gamma distribution with hyperparameters:

$$\begin{aligned} \lambda' &= \lambda + n & \nu' &= \frac{\lambda}{\lambda + n}\nu + \frac{n}{\lambda + n}\hat{\mu} \\ \alpha' &= \alpha + \frac{n}{2} & \beta' &= \beta + \frac{1}{2}\left(n\hat{s}^2 + \frac{\lambda n}{\lambda + n}\left(\hat{\mu} - \nu\right)^2\right) \end{aligned} \tag{27}$$

Often we are only interested in the posterior distribution of the mean; the marginal distribution on μ is a three-parameter (degrees of freedom, mean, scale) non-standardized t-distribution:

$$\mathbb{P}\left(l \,|\, \{x\} \;;\; \lambda, \nu, \alpha, \beta\right) = \text{StudentT}\left(l \,;\, 2\alpha', \nu', \sqrt{\frac{\beta'}{\lambda'\alpha'}}\right) \tag{28}$$

where the normal-gamma hyperparameters have been updated using $\{x\}$ according to Equation 27. Prior to any observations, the hyperparameters are set to $\lambda_0 = 0, \nu_0 = 0$ (representing a noninformative prior over location) and α_0, β_0 chosen such that $\frac{\beta_0}{\alpha_0}$ is equal to a prior value of the variance, using α_0 to toggle the prior strength. For location, we use $\alpha_0 = 10$ and $\frac{\beta_0}{\alpha_0} = 9 \times 10^{-4}$, representing a weak prior where the location standard deviation is expected to be around 3cm.

The upshot of using a conjugate prior for location measurements is that the marginal likelihood of location observations has a closed-form expression. The posterior predictive distribution for the next location observation x' is obtained by integrating out the latent parameters l, τ:

$$\begin{aligned}
\mathbb{P}\left(x' \,\middle|\, \{x\} \;;\; \lambda, \nu, \alpha, \beta\right) &= \int_{(l,\tau)} \mathbb{P}\left(x \,|\, l, \tau\right) \mathbb{P}\left(l, \tau \,|\, \{x\} \;;\; \nu, \lambda, \alpha, \beta\right) \\
&= \int_{(l,\tau)} \mathcal{N}\left(x \,;\, l, \tau^{-1}\right) \text{ NormalGamma}\left(l, \tau \,;\, \nu^-, \lambda^-, \alpha^-, \beta^-\right) \\
&= \frac{1}{\sqrt{2\pi}} \frac{\beta^{-\alpha^-}}{\beta^{+\alpha^+}} \frac{\sqrt{\lambda^-}}{\sqrt{\lambda^+}} \frac{\Gamma(\alpha^+)}{\Gamma(\alpha^-)}
\end{aligned} \tag{29}$$

where hyperparameters with "$-$" superscripts are updated according to Equation 27 using the empirical statistics of $\{x\}$ only (excluding x'), and ones with "$+$" superscripts are likewise updated but including x'. The ratio in Equation 29 assesses the fit of x' with the existing observations $\{x\}$ associated with the light.

References

A. Aldoma, F. Tombari, J. Prankl, A. Richtsfeld, L. Di Stefano, and M. Vincze. Multimodal cue integration through hypotheses verification for RGB-D object recognition and 6DOF pose estimation. In *IEEE International Conference on on Robotics and Automation*, 2013.

R. Anati, D. Scaramuzza, K.G. Derpanis, and K. Daniilidis. Robot localization using soft object detection. In *IEEE International Conference on Robotics and Automation*, 2012.

C.E. Antoniak. Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174, 1974.

N. Atanasov, B. Sankaran, J. Le Ny, T. Koletschka, G. Pappas, and K. Daniilidis. Hypothesis testing framework for active object detection. In *International Conference on Robotics and Automation*, 2013.

Y. Bar-Shalom and T.E. Fortmann. *Tracking and Data Association*. Academic Press, 1988.

J.M. Bernardo and A.F.M. Smith. *Bayesian Theory*. John Wiley, 1994.

N. Blodow, D. Jain, Z.-C. Marton, and M. Beetz. Perception and probabilistic anchoring for dynamic world state logging. In *IEEE-RAS International Conference on Humanoid Robots*, 2010.

I.J. Cox and S.L. Hingorani. An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(2):138–150, 1996.

I.J. Cox and J.J. Leonard. Modeling a dynamic environment using a Bayesian multiple hypothesis approach. *Artificial Intelligence*, 66(2):311–344, 1994.

J.L. Crowley. Dynamic world modeling for an intelligent mobile robot using a rotating ultra-sonic ranging device. In *IEEE International Conference on Robotics and Automation*, 1985.

F. Dellaert. *Monte Carlo EM for Data-Association and its Applications in Computer Vision.* PhD thesis, Robotics Institute, Carnegie Mellon University, 2001.

F. Dellaert, S.M. Seitz, C.E. Thorpe, and S. Thrun. EM, MCMC, and chain flipping for structure from motion with unknown correspondence. *Machine Learning*, 50(1–2):45–71, 2003.

R. Eidenberger and J. Scharinger. Active perception and scene modeling by planning with probabilistic 6D object poses. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010.

J. Elfring, S. van den Dries, M.J.G. van de Molengraft, and M. Steinbuch. Semantic world modeling using probabilistic multiple hypothesis anchoring. *Robotics and Autonomous Systems*, 61(2):95–105, 2013.

J. Glover and S. Popovic. Bingham Procrustean alignment for object detection in clutter. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013.

J. Glover, R.B. Rusu, and G. Bradski. Monte Carlo pose estimation with quaternion kernels and the Bingham distribution. In *Robotics: Science and Systems*, 2011.

G.D. Hager and B. Wegbreit. Scene parsing using a prior world model. *International Journal of Robotics Research*, 30(12):1477–1507, 2011.

B. Kulis and M.I. Jordan. Revisiting k-means: New algorithms via Bayesian nonparametrics. In *International Conference on Machine Learning*, 2012.

T. Kurien. Issues in the design of practical multitarget tracking algorithms. In Y. Bar-Shalom, editor, *Multitarget-Multisensor Tracking: Advanced Applications*, pages 43–84. Artech House, 1990.

K. Lai, L. Bo, X. Ren, and D. Fox. Detection-based object labeling in 3D scenes. In *IEEE International Conference on on Robotics and Automation*, 2012.

R.P.S. Mahler. *Statistical Multisource-Multitarget Information Fusion.* Artech House, 2007.

Z.-C. Marton, F. Balint-Benczedi, O.M. Mozos, N. Blodow, A. Kanezaki, L.C. Goron, D. Pangercic, and M. Beetz. Part-based geometric categorization and object reconstruction in cluttered table-top scenes. *Journal of Intelligent & Robotic Systems*, pages 1–22, 2014.

J. Mason and B. Marthi. An object-based semantic world model for long-term change detection and semantic querying. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.

R.M. Neal. Markov chain sampling methods for Dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 9(2):249–265, 2000.

R.A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A.J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. In *IEEE International Symposium on Mixed and Augmented Reality*, 2011.

A. Nüchter and J. Hertzberg. Towards semantic maps for mobile robots. *Robotics and Autonomous Systems*, 56(11):915–926, 2008.

S. Oh, S. Russell, and S. Sastry. Markov chain Monte Carlo data association for multi-target tracking. *IEEE Transactions on Automatic Control*, 54(3):481–497, 2009.

H. Pasula, S. Russell, M. Ostland, and Y. Ritov. Tracking many objects with many sensors. In *International Joint Conference on Artificial Intelligence*, 1999.

A. Pronobis and P. Jensfelt. Large-scale semantic mapping and reasoning with heterogeneous modalities. In *IEEE International Conference on Robotics and Automation*, 2012.

A. Ranganathan and F. Dellaert. Semantic modeling of places using objects. In *Robotics: Science and Systems*, 2007.

D.B. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, 1979.

R.B. Rusu, G. Bradski, R. Thibaux, and J. Hsu. Fast 3D recognition and pose using the viewpoint feature histogram. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010.

J. Sethuraman. A constructive definition of Dirichlet priors. *Statistical Sinica*, 4:639–650, 1994.

Y.W. Teh. Dirichlet processes. In C. Sammut and G.I. Webb, editors, *Encyclopedia of Machine Learning*, pages 280–287. Springer, 2010.

S. Vasudevan, S. Gächter, V. Nguyen, and R. Siegwart. Cognitive maps for mobile robots – an object based approach. *Robotics and Autonomous Systems*, 55(5):359–371, 2007.

J. Velez, G. Hemann, A.S. Huang, I. Posner, and N. Roy. Modelling observation correlations for active exploration and robust object detection. *Journal of Artificial Intelligence Research*, 44: 423–453, 2012.

H. Zender, O.M. Mozos, P. Jensfelt, G.-J.M. Kruijff, and W. Burgard. Conceptual spatial representations for indoor mobile robots. *Robotics and Autonomous Systems*, 56(6):493–502, 2008.