

Not Seeing is Also Believing: Combining Object and Metric Spatial Information

Lawson L.S. Wong, Leslie Pack Kaelbling, and Tomás Lozano-Pérez

Abstract—Spatial representations are fundamental to mobile robots operating in uncertain environments. Two frequently-used representations are occupancy grid maps, which only model metric information, and object-based world models, which only model object attributes. Many tasks represent space in just one of these two ways; however, because objects must be physically grounded in metric space, these two distinct layers of representation are fundamentally linked. We develop an approach that maintains these two sources of spatial information separately, and combines them on demand. We illustrate the utility and necessity of combining such information through applying our approach to a collection of motivating examples.

I. INTRODUCTION

Spatial representations are fundamental to mobile robots operating in uncertain environments. A navigating mobile robot needs to know which places are free to move into and what obstacles it might collide with. A mobile manipulation robot cooking at home needs to be able to find and detect objects such as kitchen utensils and ingredients. These two tasks typically represent space in distinct ways: navigation with occupancy grid maps, which we will refer to as ‘metric-level’; mobile manipulation with objects and their attributes, which is ‘object-level’. Many tasks represent space in just one of these two ways, use them in parallel without information flow, or infer one solely from the other, but rarely is there any interaction between the two levels.

Consider a motivating example, as depicted in Fig. 1. Here, a mobile robot with a camera mounted on top takes an image and sees the side of a shelf on a table. From the camera point cloud, it infers that a shelf of some known or measured size is present, and estimates the shelf’s pose, shown in red and indicated by the white arrow. Even though most of the shelf lies within an unobserved region of space, as indicated by the gray ‘fog’ on the right, the robot can infer that the space overlapping with the box at its estimated pose is occupied (by the shelf). This is an example of object-to-metric inference. Through the act of seeing the shelf, the robot also knows that the rays between its camera and the front of the shelf passed through free (unoccupied) space. Since this space is free, the robot can also infer that no objects are present in

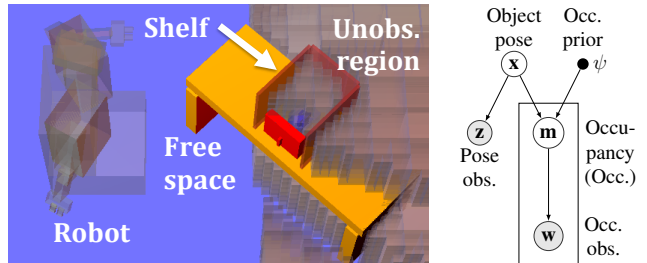


Fig. 1. A mobile robot uses object detections to infer regions of occupied space, and uses free space observations to eliminate possible locations of objects. Our framework allows inference across representational layers as depicted by the graphical model; please see Secs. III–V for details.

this space. This is an example of metric-to-object inference. We will consider more examples of both types of information interaction in this paper.

With effort, it is typically possible to use only a single layer of spatial representation. However, this can unnecessarily complicate the storage of information and the updating of the representation, because certain types of information from sensors are more compatible with specific types of representation. An identified rigid object is inherently atomic, but this is not respected when treated as a collection of discretized grid cells. If the object is moved, then instead of simply updating a ‘pose’ attribute in the object state, the entire collection of grid cells will need to be updated. Conversely, free space is easy to represent in an occupancy grid. However, because it provides information about the absence of any object, which amounts to ‘cutting holes’ in each object’s pose distribution, forcing free space information to be kept in pose space leads to complicated pose distributions and updates that scale with the number of known objects instead of the number of newly observed cells. Moreover, much of this complex updating is wasted, because information about a local region of free space would not affect an object’s pose unless the object is nearby.

Our goal is to combine the advantages of each layer of representation and provide a framework for integrating both types of information. In particular, we adopt the philosophy of keeping each type of information in its ‘natural’ representation, where it can be easily updated, and *only* combining them when queries about specific states are made. This is an efficiency trade-off between filtering and querying; we strive for simplicity and compactness in the former by delaying computation to query-time. The specific representational choices made will be explored in greater detail in Sec. III.

To illustrate our strategy, Sec. IV develops, in detail, the approach for a concrete one-dimensional discrete world

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.

Computer Science and Artificial Intelligence Laboratory,
Massachusetts Institute of Technology, Cambridge, MA 02139
{lsw,lpk,tlp}@csail.mit.edu

involving a single object. The general case is in fact not too different, and will be covered in Sec. V. Several example applications of the framework are presented in Sec. VI, where we will demonstrate, for example, how free space information can be used to reduce uncertainty in object type and pose, and why object-level representations are necessary to maintain an accurate metric spatial representation.

II. RELATED WORK

Since Moravec and Elfes [1] pioneered the occupancy grid model of space, occupancy grids have been used extensively in robotics, most notably in mapping. These maps have paved the way for tasks such as navigation and motion planning, in which knowledge of free and occupied spaces is sufficient for success. However, as we move to tasks that require richer interaction with the world, such as locating and manipulating objects, occupancy information alone is insufficient.

In the mapping community, there has been recognition that using metric representations only is insufficient. In particular, the rise of topological mapping, and the combination of the two in hybrid metric-topological mapping ([2]) suggests the utility of going beyond metric representations. These hybrid representations have been successfully applied in tasks such as navigation ([3]). A related field that has been growing recently is semantic mapping (e.g., [4], [5], [6], [7]), where typically the focus is to endow topological regions of space with semantic attributes, such as in the task of place classification. Topological and semantic information is typically extracted from metric layers (occupancy grids).

Some works in semantic mapping do place greater emphasis on the detailed modeling of objects (e.g., [8], [9], [10]). However, as with the hybrid mapping community, object-based information is rarely propagated back down to the metric level. The importance of objects is underscored by the existence of large computer vision communities that are dedicated to detecting and recognizing objects. Sophisticated methods (e.g., [11], [12], [13]) exist to build and maintain world models, which are representations of space in terms of objects and their attributes. Although vision techniques typically rely on geometric features to infer object existence, we are not aware of any method that allows for information to flow in the reverse direction, as we do in this paper.

III. PROBLEM DEFINITION AND SOLUTION STRATEGY

Consider a well-localized robot making observations in a world containing stationary objects. Since the contents of a spatial representation is ultimately a state estimate, we first describe the state. We assume that each object obj^i is described by a fixed set of attributes of interest, whose values are concatenated into a vector \mathbf{x}^i . Likewise, the world is discretized into a metric grid (not necessarily evenly spaced), where each cell $cell^j$ is endowed with another set of attributes with value \mathbf{m}^j . For concreteness, it may help to consider \mathbf{x}^i being the object pose (assuming we know which object it is), and \mathbf{m}^j being the binary occupancy value for $cell^j$. We shall explore this case further in Sec. IV, and subsequently generalize to other attributes in Sec. V.

The objects' states $\{\mathbf{x}^i\}$ and the cells' states $\{\mathbf{m}^j\}$ are not known, and are imperfectly sensed by the robot. We assume that the perception framework returns two independent types of observations, $\{\mathbf{z}_{1:Z}^i\}$ and $\{\mathbf{w}_{1:W}^j\}$, revealing information about the objects and cells respectively. The subscripts indicate that each object/cell may have multiple observations. Observations may be raw sensor readings or be the output of some intermediate perception pipeline. For example, \mathbf{w}^j may be range sensor readings, whereas \mathbf{z}^i may be the output of an object detection and pose estimation pipeline.

For convenience, we will use the following shorthand in the rest of the paper. As in the above presentation, superscripts always refer to the index of the object/cell. To avoid the clutter of set notation, we will denote the set of all objects' states, $\{\mathbf{x}^i\}$, by \mathbf{x}^\bullet ; specific indices will denote individual states (e.g., \mathbf{x}^i is obj^i 's state). Similarly, \mathbf{m}^j is $cell^j$'s state, whereas \mathbf{m}^\bullet refers to the states of all cells (previously $\{\mathbf{m}^j\}$). Likewise, for observations, \mathbf{z}_k^i is the k 'th observation associated with obj^i , \mathbf{z}_\bullet^i is the set of observations associated with obj^i , and \mathbf{z}^\bullet is the set of all object observations (previously $\{\mathbf{z}_{1:Z}^i\}$).

Our goal is to estimate the marginal posterior distributions:

$$\mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet, \mathbf{w}^\bullet) \quad \text{and} \quad \mathbb{P}(\mathbf{m}^\bullet | \mathbf{z}^\bullet, \mathbf{w}^\bullet). \quad (1)$$

In most of our examples, such as the object-pose/cell-occupancy one described above, \mathbf{x}^\bullet and \mathbf{m}^\bullet are dependent: given that an object is in pose x , the cells that overlap with the object at pose x must be occupied. Such object-based dependencies also tend to be local to the space that the object occupies and hence very non-uniform: cells that do *not* overlap with the object are essentially unaffected. The lack of uniformity dashes all hopes of a nice parametric update to the objects' states. For example, if m^j is known to be free, all poses that overlap m^j must have zero probability, thereby creating a 'hole' in pose space that is impossible to represent using a typical Gaussian pose distribution.

As a result, we must resort to non-parametric representations, such as a collection of samples, to achieve good approximations to the posterior distributions. However, the dimension of the joint state grows with the number of objects and the size of the world, and sampling in the joint state quickly becomes intractable in any realistic environment. This approach can be made feasible with aggressive factoring of the state space; however, combining different factors correctly simply introduces another fusion problem. Filtering a collection of samples over time, or particle filtering ([14], [15]), also introduces particle-set maintenance issues.

Instead of filtering in the joint state and handling complex dependencies, our strategy is to filter *separately* in the object and metric spaces, and *merge* them on demand as queries about either posterior are made. Our philosophy is to trade off filter accuracy for runtime efficiency (by using more restrictive representations that each require ignoring different parts of the perceived data), while ensuring that appropriate corrections are made when answering queries. By making typical independence assumptions within each layer, we can leverage standard representations such as a Kalman filter (for

object pose) and an occupancy grid (for metric occupancy) to make filtering efficient. Specifically, we propose to maintain the following distributions in two filters:

$$\mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet) \quad \text{and} \quad \mathbb{P}(\mathbf{m}^\bullet | \mathbf{w}^\bullet), \quad (2)$$

and *only* incorporate the other source of information at query time. Computing the posteriors in Eqn. 1 from the filtered distributions in Eqn. 2 is the subject of the next section.

IV. THE ONE-DIMENSIONAL, SINGLE-OBJECT CASE

To ground our discussion of the solution strategy, in this section we consider a simple instance of the general problem discussed in the previous section. In particular, we focus on the case of estimating the (discrete) location of a single static object and the occupancy of grid cells in a discretized one-dimensional world. The general problem involving more objects and other attributes is addressed in Sec. V.

A. Formulation

The single-object, 1-D instance is defined as follows:

- The 1-D world consists of C contiguous, unit-width cells with indices $1 \leq j \leq C$.
- A static object of interest, with known length L , exists in the world. Its location, the lowest cell index it occupies, is the only attribute being estimated. Hence its state \mathbf{x} satisfies $\mathbf{x} \in [1, C - L + 1] \triangleq \{1, \dots, C - L + 1\}$.
- We are also interested in estimating the occupancy of each cell $cell^j$. Each cell's state \mathbf{m}^j is binary, with value 1 if it is occupied and 0 if it is free.
- Cells may be occupied by the object, occupied by 'dirt'/'stuff', or be free. 'Stuff' refers to physically-existing entities that we either cannot yet or choose not to identify. Imagine only seeing the tip of a handle (which makes the object difficult to identify) or, as the name suggests, a ball of dirt (which we choose to ignore except note its presence). We will not explicitly distinguish between the two types of occupancy; the cell's state has value 1 if it is occupied by either the object or 'stuff', and 0 if it is free.
- The assumption above, that cells can be occupied by non-object entities, allows us to ascribe a simple prior model of occupancy: each cell is occupied independently with known probability $\mathbb{P}(\mathbf{m}^j = 1) = \psi$. This prior model and cell independence assumption are commonly used in the occupancy grid literature (see, e.g., [16]). This may be inaccurate, especially if the object is long and ψ is small.
- Noisy observations \mathbf{z}_\bullet of the single object's location \mathbf{x} and observations \mathbf{w}^\bullet of the cells' occupancies \mathbf{m}^\bullet are made. We will be intentionally agnostic to the specific sensor model used, and only assume that appropriate filters are used in light of the noise models.
- The object and metric filters maintain $\mathbb{P}(\mathbf{x} | \mathbf{z}_\bullet)$ and $\mathbb{P}(\mathbf{m}^\bullet | \mathbf{w}^\bullet)$ respectively. We assume that the former is a discrete distribution over the domain of \mathbf{x} , and the latter is an occupancy grid, using the standard log-odds ratio $\ell^j = \log \frac{\mathbb{P}(\mathbf{m}^j=1 | \mathbf{w}_\bullet^j)}{\mathbb{P}(\mathbf{m}^j=0 | \mathbf{w}_\bullet^j)}$ for each cell's occupancy.
- States of distinct cells are assumed to be conditionally independent given the object state \mathbf{x} . This is a relaxation

of the assumption cells are independent, which is typically assumed in occupancy grids. The current assumption disallows arbitrary dependencies between cells; only dependencies mediated by objects are allowed. For example, two adjacent cells may be occupied by the same object and hence are dependent if the object's location is not known.

As mentioned in the previous section, what makes this problem interesting is that \mathbf{x} and \mathbf{m}^\bullet are dependent. In this case, the crucial link is that an object that is located at \mathbf{x} *necessarily occupies* cells with indices $j \in \mathcal{J}(\mathbf{x}) \triangleq [\mathbf{x}, \mathbf{x} + L - 1]$, and therefore these cells must have as state $\mathbf{m}^j = 1$. This means that states of a subset of cells are strongly dependent on the object state, and we expect this to appear in the metric posterior $\mathbb{P}(\mathbf{m}^\bullet | \mathbf{z}_\bullet, \mathbf{w}^\bullet)$. Likewise, occupancy/freeness of a cell also supports/opposes respectively the hypothesis that an object overlaps the cell. However, the latter dependency is weaker than the former one, as an occupied cell can be due to 'dirt' (or other objects, though not in this case), and a free cell typically only eliminates a small portion of the object location hypotheses.

B. Cell occupancy posterior

We now use this link between \mathbf{x} and \mathbf{m}^\bullet to derive the desired posterior distributions from Eqn. 1. We first consider the posterior occupancy \mathbf{m}^j of a single cell $cell^j$. Intuitively, we expect that if the object likely overlaps $cell^j$, the posterior occupancy should be close to 1, whereas if the object is unlikely to overlap the cell, then the posterior occupancy should be dictated by the 'stuff' prior and relevant occupancy observations (\mathbf{w}_\bullet^j). Since we do not know the exact location of the object, we instead have to consider all possibilities:

$$\mathbb{P}(\mathbf{m}^j | \mathbf{z}_\bullet, \mathbf{w}^\bullet) = \sum_{\mathbf{x}} \mathbb{P}(\mathbf{m}^j | \mathbf{x}, \mathbf{w}_\bullet^j) \mathbb{P}(\mathbf{x} | \mathbf{z}_\bullet, \mathbf{w}^\bullet). \quad (3)$$

In the first term, because \mathbf{x} is now explicitly considered, object observations \mathbf{z}_\bullet are no longer informative and are dropped. Since we assumed that cells are conditionally independent given the object state, all other cells' observations are dropped too. The second term is the posterior distribution on the object location, which will be discussed later.

The term $\mathbb{P}(\mathbf{m}^j | \mathbf{x}, \mathbf{w}_\bullet^j)$ can be decomposed further:

$$\mathbb{P}(\mathbf{m}^j | \mathbf{x}, \mathbf{w}_\bullet^j) \propto \mathbb{P}(\mathbf{w}_\bullet^j | \mathbf{m}^j) \mathbb{P}(\mathbf{m}^j | \mathbf{x}) \quad (4)$$

The second term, $\mathbb{P}(\mathbf{m}^j | \mathbf{x})$, serves as the link between cells and objects. By the discussion above, for $j \in \mathcal{J}(\mathbf{x})$, i.e., cells that the object at location \mathbf{x} overlaps, \mathbf{m}^j must be 1. In this case, Eqn. 4 is only non-zero for $\mathbf{m}^j = 1$, so $\mathbb{P}(\mathbf{m}^j = 1 | \mathbf{x}, \mathbf{w}_\bullet^j)$ must also be 1. For $j \notin \mathcal{J}(\mathbf{x})$, the cell is unaffected by the object, hence $\mathbb{P}(\mathbf{m}^j | \mathbf{x}) = \mathbb{P}(\mathbf{m}^j)$. Eqn. 4 in this case is, by reverse application of Bayes' rule, proportional to $\mathbb{P}(\mathbf{m}^j | \mathbf{w}_\bullet^j)$, and since this is in fact a distribution, $\mathbb{P}(\mathbf{m}^j | \mathbf{x}, \mathbf{w}_\bullet^j) = \mathbb{P}(\mathbf{m}^j | \mathbf{w}_\bullet^j)$. This reflects that for $j \notin \mathcal{J}(\mathbf{x})$, the cell's state is independent of the object state. In summary:

$$\mathbb{P}(\mathbf{m}^j | \mathbf{x}, \mathbf{w}_\bullet^j) = \begin{cases} 1 & \text{if } j \in \mathcal{J}(\mathbf{x}), \\ \mathbb{P}(\mathbf{m}^j | \mathbf{w}_\bullet^j) & \text{otherwise.} \end{cases} \quad (5)$$

This ‘link’ between object and cell states matches the intuition given above: the cell is necessarily occupied if the object overlaps it; otherwise, the object state is ignored and only occupancy observations are used. The probability value $\mathbb{P}(\mathbf{m}^j | \mathbf{w}^j)$ is readily available from the metric filter (for an occupancy grid with log-odds ratio ℓ^j for $cell^j$, the desired probability is $1 - \frac{1}{1 + \exp(\ell^j)}$). Combining Eqns. 3 and 5 results in a nicely interpretable posterior:

$$\mathbb{P}(\mathbf{m}^j | \mathbf{z}_\bullet, \mathbf{w}^\bullet) = p_{\text{overlap}} + \mathbb{P}(\mathbf{m}^j | \mathbf{w}^j) (1 - p_{\text{overlap}}), \quad (6)$$

where $p_{\text{overlap}} \triangleq \mathbb{P}(\mathbf{x} \in [j - L + 1, j] | \mathbf{z}_\bullet, \mathbf{w}^\bullet)$, the posterior probability that the object is in a location that overlaps $cell^j$. To compute this value, we need the object location’s posterior distribution, which we turn to now.

C. Object location posterior

By Bayes’ rule,

$$\mathbb{P}(\mathbf{x} | \mathbf{z}_\bullet, \mathbf{w}^\bullet) \propto \mathbb{P}(\mathbf{w}^\bullet | \mathbf{x}, \mathbf{z}_\bullet) \mathbb{P}(\mathbf{x} | \mathbf{z}_\bullet) = \mathbb{P}(\mathbf{w}^\bullet | \mathbf{x}) \mathbb{P}(\mathbf{x} | \mathbf{z}_\bullet). \quad (7)$$

The second term is maintained by the object filter, and in this context acts as the ‘prior’ of the object location given only object-level observations. This distribution is adjusted by the first term, which weighs in the likelihood of occupancy observations. To evaluate this, we need to consider the latent cell occupancies \mathbf{m}^\bullet , and the constraint imposed by \mathbf{x} . Once again, cells overlapping the object must be occupied ($\mathbf{m}^j = 1$), so we only need to consider possibilities for the other cells. The non-overlapping cells are independent of \mathbf{x} , and are occupied according to the prior model (independently with probability ψ). Hence:

$$\begin{aligned} \mathbb{P}(\mathbf{w}^\bullet | \mathbf{x}) &= \sum_{\mathbf{m}^\bullet} \mathbb{P}(\mathbf{w}^\bullet, \mathbf{m}^\bullet | \mathbf{x}) \\ &= \left[\prod_{j \notin \mathcal{J}(\mathbf{x})} \sum_{\mathbf{m}^j=0}^1 \mathbb{P}(\mathbf{w}^j | \mathbf{m}^j) \mathbb{P}(\mathbf{m}^j) \right] \left[\prod_{j \in \mathcal{J}(\mathbf{x})} \mathbb{P}(\mathbf{w}^j | \mathbf{m}^j = 1) \right] \\ &= \left[\prod_j \mathbb{P}(\mathbf{w}^j) \right] \left[\prod_{j \notin \mathcal{J}(\mathbf{x})} \sum_{\mathbf{m}^j} \mathbb{P}(\mathbf{m}^j | \mathbf{w}^j) \right] \left[\prod_{j \in \mathcal{J}(\mathbf{x})} \frac{\mathbb{P}(\mathbf{m}^j = 1 | \mathbf{w}^j)}{\mathbb{P}(\mathbf{m}^j = 1)} \right] \\ &= \eta(\mathbf{w}^\bullet) \times 1 \times \left[\prod_{j \in \mathcal{J}(\mathbf{x})} \frac{1}{\psi} \left(1 - \frac{1}{1 + \exp(\ell^j)} \right) \right] \quad (8) \end{aligned}$$

where in the second line we utilized the conditional independence of cell states given \mathbf{x} to factor the expression, and $\eta(\mathbf{w}^\bullet)$ represents the first product in the penultimate line.

When substituting Eqn. 8 back into Eqn. 7, recall that since \mathbf{w}^\bullet is given, and we only need $\mathbb{P}(\mathbf{w}^\bullet | \mathbf{x})$ up to proportionality, we can ignore the $\eta(\mathbf{w}^\bullet)$ term. Hence:

$$\mathbb{P}(\mathbf{x} | \mathbf{z}_\bullet, \mathbf{w}^\bullet) \propto \left[\prod_{j \in \mathcal{J}(\mathbf{x})} \frac{1}{\psi} \left(1 - \frac{1}{1 + \exp(\ell^j)} \right) \right] \mathbb{P}(\mathbf{x} | \mathbf{z}_\bullet). \quad (9)$$

Note that the expression above only contains $O(L)$ terms, since $\mathcal{J}(x)$ contains exactly L cells. The complexity therefore scales with the number of cells the object affects, instead of with the whole world (containing C cells, which is potentially much greater than L). For discrete \mathbf{x} with X

possible states, computing $\mathbb{P}(\mathbf{x} | \mathbf{z}_\bullet, \mathbf{w}^\bullet)$ therefore requires $O(LX)$ time, since Eqn. 7 must be normalized over all possible \mathbf{x} . Finally, we have all the pieces needed to compute $\mathbb{P}(\mathbf{m}^j | \mathbf{z}_\bullet, \mathbf{w}^\bullet)$ as well using Eqn. 6. To compute both the object and metric posterior distributions, we first find the former using Eqn. 9, then find the posterior occupancy of each cell using Eqn. 6. This procedure requires $O(LX + C)$ time. In practice, when operating in local regions of large worlds, it is unlikely that one would want the posterior occupancy of all cells in the world; only cells of interest need to have their posterior state computed.

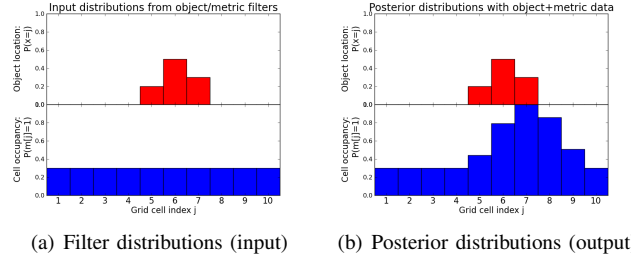


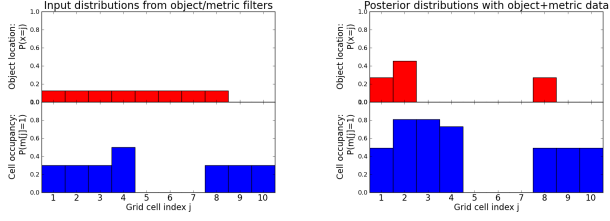
Fig. 2. Using only object observations, the object filter maintains a distribution over the object’s locations (top left). The object filter contains a *single* distribution, so top plots each sum to 1, whereas the metric filter contains a *collection* of binary distributions, one for each cell, so bottom plots do *not* sum to 1. Some cells have increased posterior probability of occupancy (bottom right), even though *no occupancy observations have been made*. Please see text in Sec. IV-D for details.

D. Demonstrations

To illustrate the above approach, we consider two simple examples where the world contains $C = 10$ cells and a single object of length $L = 3$. In each case, only one type of information (object location or cells’ occupancy) has been observed. The methods described in this section are used to propagate the information to the other representation.

In Fig. 2, we consider the case when only object locations have been observed. Fig. 2(a) show distributions obtained from object (top) and metric (bottom) filters, i.e., $\mathbb{P}(\mathbf{x} | \mathbf{z}_\bullet)$ and $\mathbb{P}(\mathbf{m}^\bullet | \mathbf{w}^\bullet)$ respectively. Note that the object filter contains a *single* distribution, so the top plot sums to 1, whereas the metric filter contains a *collection* of binary distributions, one for each cell, so the bottom plot does *not* sum to 1. The object filter determines that the object can only be located at cells 5–7 (recall that this is the left-most point of the object). No occupancy observations have been made, so each cell’s occupancy probability is initially the prior value, $\psi = 0.3$. In Fig. 2(b) after applying our framework, the posterior occupancy distribution $\mathbb{P}(\mathbf{m}^\bullet | \mathbf{z}_\bullet, \mathbf{w}^\bullet)$ reflects the fact that cells 5–9 might be occupied by the object, even though *no occupancy measurements have been made*. In particular, all possibilities of the object location require $cell^7$ to be occupied, hence its occupancy probability is 1. Cells with no possible overlap with the object are left unchanged. The distribution on object location is unchanged, too, since there are no additional observations to be considered.

In Fig. 3, only occupancies of some cells have been observed. Cells 5–7 have many observations indicating that they are free, and cell 4 had only one observation indicating that it is occupied. No object observations have been made,



(a) Filter distributions (input) (b) Posterior distributions (output)

Fig. 3. Using only cell occupancy/freeness observations, the posterior of the object’s location is changed drastically even though *the object has never been observed*. Please see text in Sec. IV-D for details.

so the object location distribution is uniform over the feasible range. The free cells in the middle of the location posterior distribution (top right) indicate that it is highly unlikely that any object can occupy those cells (which correspond to $\mathbf{x} \in [3, 7]$). This makes the posterior distribution multi-modal. Also, the weak evidence that *cell*⁴ is occupied gives a slight preference for $\mathbf{x} = 2$. Again, even though *the object has never been observed*, the posterior distribution on its location is drastically narrowed! Unlike the previous case, the occupancy distribution has changed, too, by virtue of the domain assumption that an object must exist. Unobserved cells are also affected by this process; in fact, *cell*² and *cell*³ are now even *more* likely to be occupied than *cell*⁴ (which had the only observation of being occupied) because of the possibility that $\mathbf{x} = 1$.

V. GENERALIZING TO ARBITRARY STATES

The previous section used several concrete simplifications: the world was one-dimensional, exactly one object existed in the world, the object’s shape (length) was given, and the only attributes considered were object location and cell occupancy. We will remove all these simplifications in this section. We will also discuss a way of handling continuous object states at the end.

Despite removing many simplifications, the conceptual framework for computing the two desired posterior distributions is actually quite similar to the development in the previous section. The major differences now are that multiple objects are present ($\mathbf{x}^\bullet, \mathbf{z}^\bullet$ instead of \mathbf{x}, \mathbf{z}), and that domain-specific derivations are no longer applicable in general. We still require the core representational assumption that an object-based filter and a metric-based filter are maintained to provide efficient access to $\mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet)$ and $\mathbb{P}(\mathbf{m}^\bullet | \mathbf{w}^\bullet)$ respectively. The latter will typically be maintained independently for each cell, with distribution $\mathbb{P}(\mathbf{m}^j | \mathbf{w}_\bullet^j)$ for *cell*^j. The typical grid cell assumptions are retained as well: cell states are conditionally independent given all object states, and states have a known prior distribution $\mathbb{P}(\mathbf{m}^j)$.

Following the derivation in Eqns. 3 and 4, we get for *cell*^j’s posterior distribution:

$$\mathbb{P}(\mathbf{m}^j | \mathbf{z}^\bullet, \mathbf{w}^\bullet) = \sum_{\mathbf{x}^\bullet} \mathbb{P}(\mathbf{m}^j | \mathbf{x}^\bullet, \mathbf{w}_\bullet^j) \mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet, \mathbf{w}^\bullet), \text{ where} \quad (10)$$

$$\mathbb{P}(\mathbf{m}^j | \mathbf{x}^\bullet, \mathbf{w}_\bullet^j) \propto \mathbb{P}(\mathbf{w}_\bullet^j | \mathbf{m}^j) \mathbb{P}(\mathbf{m}^j | \mathbf{x}^\bullet) \propto \frac{\mathbb{P}(\mathbf{m}^j | \mathbf{w}_\bullet^j)}{\mathbb{P}(\mathbf{m}^j)} \mathbb{P}(\mathbf{m}^j | \mathbf{x}^\bullet). \quad (11)$$

Again assuming that we have already computed the posterior object state $\mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet, \mathbf{w}^\bullet)$, all other terms are given except for $\mathbb{P}(\mathbf{m}^j | \mathbf{x}^\bullet)$. This distribution is the fundamental link between cells and objects, specifying in a generative fashion how objects’ states affect each cell’s state (which can be considered individually since cell states are conditionally independent given \mathbf{x}^\bullet). We will see other examples of this linking distribution in the next section.

For the posterior distribution on object states, we can likewise follow the derivation in Eqns. 7 and 8:

$$\begin{aligned} \mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet, \mathbf{w}^\bullet) &\propto \mathbb{P}(\mathbf{w}^\bullet | \mathbf{x}^\bullet) \mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet), \text{ where} & (12) \\ \mathbb{P}(\mathbf{w}^\bullet | \mathbf{x}^\bullet) &= \sum_{\mathbf{m}^\bullet} \mathbb{P}(\mathbf{w}^\bullet | \mathbf{m}^\bullet) \mathbb{P}(\mathbf{m}^\bullet | \mathbf{x}^\bullet) \\ &= \sum_{\mathbf{m}^\bullet} \left[\prod_j \mathbb{P}(\mathbf{w}_\bullet^j | \mathbf{m}^j) \right] \left[\prod_j \mathbb{P}(\mathbf{m}^j | \mathbf{x}^\bullet) \right] \\ &\propto \prod_j \left[\sum_{\mathbf{m}^j} \frac{\mathbb{P}(\mathbf{m}^j | \mathbf{w}_\bullet^j)}{\mathbb{P}(\mathbf{m}^j)} \mathbb{P}(\mathbf{m}^j | \mathbf{x}^\bullet) \right]. & (13) \end{aligned}$$

Again, all terms needed to compute the above are available from the filters, the cell prior, and the object-cell link $\mathbb{P}(\mathbf{m}^j | \mathbf{x}^\bullet)$ described earlier.

As in the previous section, we can compute this latter posterior distribution more efficiently by considering only the cells that objects affect. For any particular assignment to \mathbf{x}^\bullet , let $\mathcal{J}(\mathbf{x}^\bullet)$ be defined to be the indices of cells whose state \mathbf{m}^j depends on \mathbf{x}^\bullet . This implies that if $j \notin \mathcal{J}(\mathbf{x}^\bullet)$, then $\mathbb{P}(\mathbf{m}^j | \mathbf{x}^\bullet) = \mathbb{P}(\mathbf{m}^j)$, and their respective terms in the product of Eqn. 13 are independent of \mathbf{x}^\bullet . In fact, for $j \notin \mathcal{J}(\mathbf{x}^\bullet)$, the sum is equal to 1, a consequence of the fact that $\mathbb{P}(\mathbf{w}_\bullet^j | \mathbf{x}^\bullet) = \mathbb{P}(\mathbf{w}_\bullet^j)$ in this case. Hence:

$$\begin{aligned} \mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet, \mathbf{w}^\bullet) &\propto \left[\prod_{j \in \mathcal{J}(\mathbf{x}^\bullet)} \sum_{\mathbf{m}^j} \frac{\mathbb{P}(\mathbf{m}^j | \mathbf{w}_\bullet^j)}{\mathbb{P}(\mathbf{m}^j)} \mathbb{P}(\mathbf{m}^j | \mathbf{x}^\bullet) \right] \mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet). & (14) \end{aligned}$$

Similar to Eqn. 9, the number of product terms has been reduced from the number of cells to $O(|\mathcal{J}(\mathbf{x}^\bullet)|)$, for each \mathbf{x}^\bullet . This is potentially a major reduction because objects, for each particular state they are in, may only affect a small number of cells (e.g., the ones they occupy). Unfortunately, the expression still scales with the domain size of \mathbf{x}^\bullet , which grows exponentially with the number of objects. In practice, approximations can be made by bounding the number of objects considered jointly and aggressively partitioning objects into subsets that are unlikely to interact with each other. Alternatively, sampling values of \mathbf{x}^\bullet from the filter posterior $\mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet)$ can produce good state candidates.

Object state attributes can be continuous, for example using Gaussian distributions to represent pose. However, the above framework can only handle discrete states. Apart from discretizing the state space, one can instead sample objects’ states from the filter $\mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet)$ and use Eqn. 14 to form an approximate posterior distribution, represented as a weighted collection of samples. These samples can then be used in Eqn. 10 to compute a Monte-Carlo estimate of *cell*^j’s posterior distribution $\mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet, \mathbf{w}^\bullet)$.

VI. APPLICATIONS

In this section, we will look at several scenarios where object-based and metric-based information need to be considered together. First, we will introduce additional attributes (besides location and occupancy from Sec. IV).

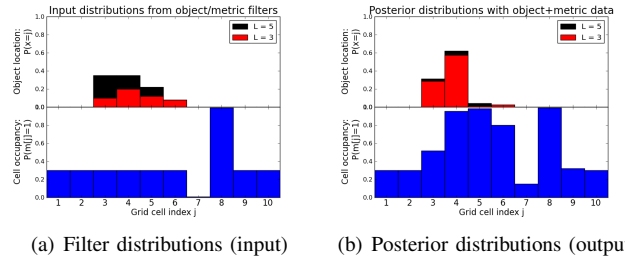
A. Shape-based object identification

When detecting and tracking objects in the world, uncertainty typically arises in more attributes than just location/pose. In particular, object recognition algorithms are prone to confusing object types, especially if we only have a limited view of the object of interest. When multiple instances of the same object type are present, we also run into data association issues. Furthermore, we may even be unsure about the number of objects in existence. Sophisticated filters (e.g., [10], [11], [12], [13]) can maintain distributions over hypotheses of the world, where a hypothesis in our context is a assignment to the joint state \mathbf{x}^\bullet .

Let us revisit the one-dimensional model of Sec. IV again, this time with uncertainty in the object type. In particular, the single object’s length L is unknown, and is treated as an attribute in \mathbf{x}^\bullet (in addition to the object’s location). Suppose that after making some observations of the object, we get $\mathbb{P}(\mathbf{x}^\bullet | \mathbf{z}^\bullet)$ from the filter, as shown in Fig. 5(a) (top). The filter has identified two possible lengths of the object ($L = 3, 5$). Here we visualize the two-dimensional distribution as a stacked histogram, where the bottom bars (red) shows the location distribution for $L = 3$, and the top bars (black) for $L = 5$. The total height of the bars is the marginal distribution of the object’s location. Suppose we have also observed that $cell^7$ is most likely empty, and $cell^8$ most likely occupied (the occupancy grid gives probability of occupancy 0.01 and 0.99 for the two cells respectively). The posterior distributions obtained by combining the filters’ distributions is shown on in Fig. 5(b).

In the object-state posterior, the main difference is that the probability mass has shifted away from the $L = 5$ object type, and towards locations on the left. Both effects are caused by the free space observations of $cell^7$. Because all locations for $L = 5$ states cause the object to overlap $cell^7$, implying that $cell^7$ is occupied, the observations that suggest otherwise cause the marginal probability of $L = 5$ to drop from 0.50 to 0.10. The drop in probability for locations 5 and 6 is due to the same reason. In conclusion, incorporating occupancy information has allowed us to reduce the uncertainty in both object location *and* object type (length).

Interestingly, among the $L = 5$ states, although location 3 had the highest probability from the object filter, it has the lowest posterior probability mass. This minor effect comes from the likely occupancy of $cell^8$, which lends more evidence to the other $L = 5$ states (which overlap $cell^8$) but not for location 3 (which does not overlap). However, the strong evidence of $cell^8$ ’s occupancy has much less of an effect compared to the free space evidence of $cell^7$. This example highlights the fundamental asymmetry in occupancy information. Recall that the prior model allows for unidentified, non-object ‘stuff’ to exist in the world, stochastically



(a) Filter distributions (input) (b) Posterior distributions (output)

Fig. 5. A 1-D scenario with a single object, but now the object’s length is uncertain as well. The object filter (top left) determines that the object may have length $L = 3$ or 5 , and for either case, may be in one of several locations. Because of a strong free space occupancy observation in $cell^7$, the uncertainty in object length has decreased significantly in the posterior object distribution (top right), because a $L = 5$ object must contradict the free space evidence of $cell^7$. Please see text in Sec. VI-A for more details.

with probability ψ . That $cell^8$ is occupied only suggests it is overlapped by some object, or contains ‘stuff’. In particular, this is the interpretation for $cell^8$ for the two most likely $L = 3$ states in the posterior. An object overlapping the cell would gain evidence, but the cell’s occupancy does not need to be explained by an object. In contrast, $cell^7$ being free means that *none* of the objects can overlap it, thereby enforcing a strong constraint on each object’s state. In the example shown in Fig. 5, this constraint allowed us to identify that the object is most likely of shorter length.

B. Physical non-interpenetration constraints

When multiple objects are present in the world, a new physical constraint appears: objects cannot interpenetrate each other ([17]). For example, in the 1-D scenario, this means that for any pair of blocks, the one on the right must have location $x^r \geq x^l + L^l$, where x^l and L^l is the location and length of the left block respectively. This is a constraint in the joint state space that couples together all object location/pose variables. One possible solution is to build in the constraint into the domain of \mathbf{x}^\bullet by explicitly disallowing joint states that violate this constraint. Although this is theoretically correct, it forces filtering to be done in the intractable joint space of all object poses, since there is in general no exact way to factor the constraint.

We now consider an alternate way to handle object non-interpenetration that is made possible by considering metric cell occupancies. So far, we have only distinguished between cells being occupied or free, but in the former case there is no indication as to what occupies the cell. In particular, the model so far allows interpenetration because two objects can occupy the same cell, and the cell state being occupied is still consistent. To disallow this, we consider expanding the occupancy attribute for grid cells. We propose splitting the previous ‘occupied’ value ($\mathbf{m}^j = 1$) into separate values, one for each object index, and one additional value for ‘stuff’/unknown. That is, the cell not only indicates that it is occupied, but also which object is occupying it (if known). Then in the object-metric link $\mathbb{P}(\mathbf{m}^j | \mathbf{x}^\bullet)$, if, for example, obj^2 overlaps $cell^j$, \mathbf{m}^j is enforced to have value 2. The non-interpenetration constraint naturally emerges, since if obj^1 and obj^2 interpenetrate, they must overlap in some cell $cell^j$,

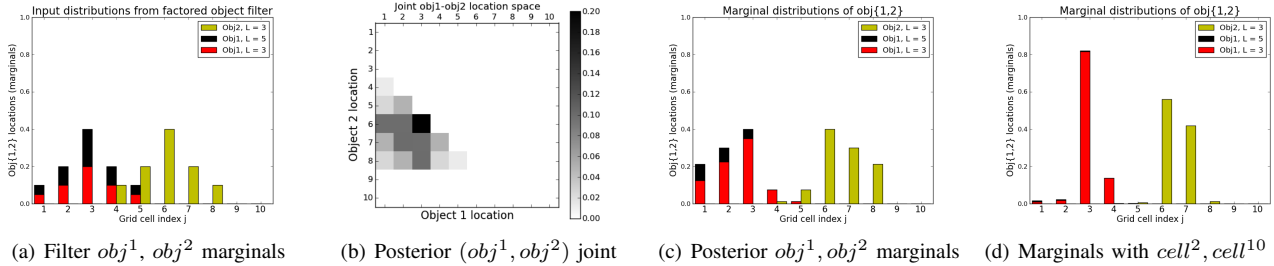


Fig. 4. A 1-D scenario with two objects. When multiple objects are present, a physical non-interpenetration constraint is introduced. (a) The filter maintains the object locations as a product of marginals, which does not respect the constraint. The red/black bars are for obj^1 locations with lengths $L = 3$ and 5 respectively; the yellow bars are for obj^2 locations. (b) After considering the constraint in metric occupancy space, the posterior joint distribution shows that the two object locations are highly coupled. (c) The posterior marginal distributions reflect the constraint’s effects: locations in the middle are unlikely either object’s left-most cell, because it forces the other object into low-probability states. (d) If additionally $cell^2$ and $cell^{10}$ are observed to likely be free, only a few joint states are possible. Also, the possibility of obj^1 having length $L = 5$ is essentially ruled out. Please see text in Sec. VI-B for details.

whose value is enforced to be both 1 and 2, a situation with zero probability. Such violating joint object states are hence naturally pruned out when evaluating the posterior (Eqn. 14). In particular, even if the object filter’s distribution contains violating states with non-zero probability, by considering the objects’ effects on grid cells the constraint is enforced and such violating states have zero probability in $\mathbb{P}(x^\bullet | z^\bullet, w^\bullet)$. We can therefore use a more efficient filter representation that ignores the constraint, such as a product of marginal location distributions, and enforce the constraint at query time when metric information is incorporated.

In our 1-D world with two objects, suppose their locations are maintained by the filter as a product of marginal distributions, as depicted in Fig. 4(a). The marginal distribution for obj^1 is shown in red/black bars; the yellow bars represent the marginal distribution for obj^2 . In addition, there is uncertainty in the length of obj^1 . Note that this also factors into the non-interpenetration constraint, since, for example, obj^1 at $x^1 = 4$ with $L = 3$ is compatible with obj^2 at $x^2 = 7$, but this is not true for obj^1 with $L = 5$ and the same locations. After enforcing the non-interpenetration constraint by reasoning about metric cell states, the posterior joint object location distribution is shown in Fig. 4(b). Here white location-pairs have zero probability, and the most likely joint state $(x^1, x^2) = (3, 6)$ has joint probability of 0.2. Based on the marginals and the constraint, obj^1 must be to the left of obj^2 , hence the only non-zero probabilities are above the diagonal. The posterior marginal distributions of the two objects’ states are depicted in Fig. 4(c). Locations in the middle are less likely for both objects since, for each object, such locations force the other object into low-probability states. Also, the length L must be 3 the two right-most obj^1 locations, otherwise it will be impossible to fit obj^2 in any locations with non-zero marginal probability.

Suppose we additionally observe that $cell^2$ and $cell^{10}$ are likely to be empty. This greatly restricts the states of the objects; the posterior marginal distributions of the two objects’ states in this case is shown in Fig. 4(d). We see that there are basically only two likely locations for obj^1 now, and that its length is most likely 3 (with probability 0.98). This is because the additional cell observations constrain both objects to be between $cell^2$ and $cell^9$, of which the only object location possibilities are $(x^1, x^2) \in$

$\{(3, 6), (3, 7), (4, 7)\}$. The larger marginal distribution of obj^1 at location 3 is due to the fact that two joint states are possible, and each has relatively high probability from the input marginal distributions given by the objects filter.

In summary, occupancy information can both enforce physical non-interpenetration constraints, as well as reduce uncertainty in object states via free space observations.

C. Demonstration on robot

We have also empirically validated our approach on a small real-world example, as shown in Fig. 6 and in the accompanying video (http://lis.csail.mit.edu/movies/ICRA14_1678_VI-fi.mp4). The initial setup is shown in Fig. 6(a): a toy train is placed on a table, and a PR2 robot is attempting to look at it. However, its view is mostly blocked by a board (Fig. 6(b)); only a small part of the train’s front is visible. A simple object instance detector recognizes it as the front of a toy train. The question is, does the train have one car (short) or two cars (long) (Figs. 6(c) and 6(d))? The true answer is one train car in this example.

One way to determine the answer is to move away the occluding board (or equivalently, moving to a better viewpoint). This is depicted by the occupancy grids in Figs. 6(e)-6(g). The grid consists of cubes with side length 2cm, within a $1m \times 0.4m \times 0.2m$ volume (hence 10^4 cubes in total). The figures show the grid projected onto the table (vertical dimension collapsed). The yellow and black points show free space and occupancy observations respectively. These observations are determined from depth images returned by a head-mounted Kinect camera: points indicate occupied cells, and rays between points and the camera contain free cells.

Since it is known that there must be a toy train with at least one car, performing object-to-metric inference results in additional cells with *inferred* potential occupancy, as shown by the blue (one car) and green (two car) cases. The number of occupied cells is greater than the train’s volume due to uncertainty in the object pose; the cells near the middle have a darker shade because they are more likely to be occupied. As the board is moved gradually to the right, more occupancy observations are collected, and eventually there are free space observations where a second train car should have occupied (circled in Fig. 6(g)). By inference similar to that from Sec. VI-A, the two-car case is therefore ruled out.

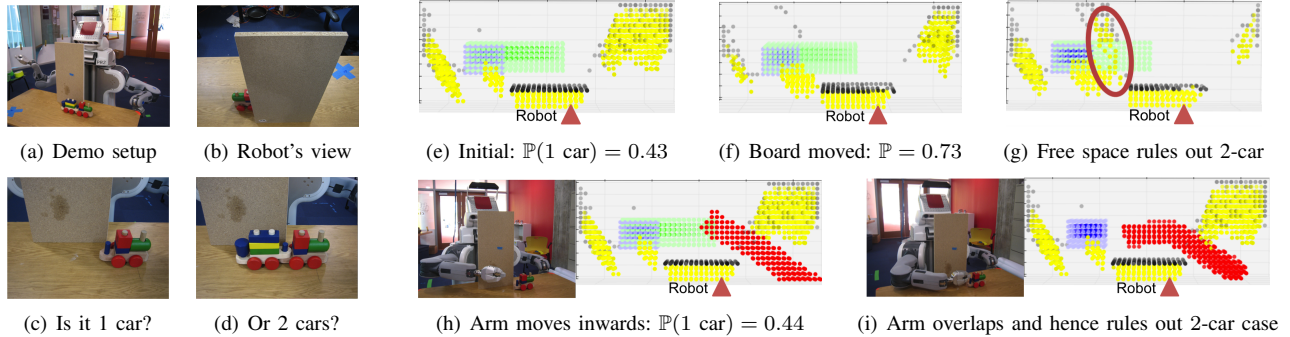


Fig. 6. A 3-D demonstration on a PR2 robot. Plots show occupancy grids with $1\text{m} \times 0.4\text{m} \times 0.2\text{m}$ volume, containing 10^4 cubes of side length 2cm, with the final (vertical) dimension projected onto the table. Colors depict occupancy type/source: Yellow = free space observation; Black = occupancy observation; Blue = inferred occupancy from one-car train; Green = inferred occupancy from two-car train; Red = occupied by robot in its current state. In this projection, the robot is situated at the bottom center of the plot, facing ‘upwards’; the black line observed near the bottom corresponds to the board. (a)-(b) A toy train is on a table, but only part of the front is visible to the robot. (c)-(d) This is indicative of two possible scenarios: the train has one car or two cars; there is in fact only one car. (e)-(g) One way to determine the answer is to move the occluding board away. This reveals free space where the second car would have been (circled in (e)), hence ruling out the two-car case. (h)-(i) Another way is to use the robot arm. If the arm successfully sweeps through cells without detecting collision, the cells must have originally been free and are now occupied by the arm. Sweeping through where the second car would have been therefore eliminates the possibility of the train being there. Please see text in Sec. VI-C and the accompanying video for details.

Without moving either the board or the viewpoint, another way to arrive at the same conclusion is to use the robot arm, shown in Figs. 6(h) and 6(i). Here, occupancy ‘observations’ (red) are derived from the robot model – cells overlapping the robot in its current configuration must be occupied by the robot. In particular, as in Sec. VI-B, we can augment the occupancy attribute to indicate that these cells are occupied by the robot. As the robot arm sweeps through the space where the second train car would have been, no collisions are detected. This indicates that the space the arm swept through is free or occupied by the robot, which by inference similar to that from Sec. VI-B rules out the two-car case.

VII. CONCLUSIONS AND FUTURE WORK

Through several examples, we demonstrated that there are many plausible situations in which representing space using both object-based and metric representations is useful and necessary. To combine object-based and metric information, instead of filtering in the complicated joint state space, we adopted a philosophy of filtering in separate, easily-manageable spaces, then only computing fused estimates on demand. The approach for combining object-level and metric-level states was developed extensively in the paper.

The given examples have been on small, low-dimensional domains. The prospects of directly scaling up the presented approach are unclear. As discussed in Sec. IV-C, the complexity of the generic inference calculation is $O(LX + C)$, where L is the number of cells objects occupy, X is the number of (discrete) attribute settings for all objects, and C is the number of grid cells in the world. Potential efficiencies may be exploited if X is (approximately) factored or if adaptive grids such as octrees are used. Nevertheless, the number of objects and cells needed to represent large spatial environments will still present challenges. Instead, our approach is perhaps most useful for *fine local* estimation: information fusion is *only* performed for few objects/attributes and small areas of great interest (e.g., to a given task), in cases where information from either the object-level or metric-level representation alone is insufficient.

More theoretical and empirical work is needed to determine the ramifications of our representation when used in large environments over long periods of time. Handling continuous and high-dimensional state (attribute) spaces, as well as scaling up to larger environments containing many objects, are subjects of future work. Nevertheless, even in its current simplistic and generic form, our approach enables novel lines of spatial inference that could not be accomplished using single layers of spatial representation.

REFERENCES

- [1] H. Moravec and A. E. Elfes, “High resolution maps from wide angle sonar,” in *ICRA*, 1985.
- [2] S. Thrun, “Learning metric-topological maps for indoor mobile robot navigation,” *Artificial Intelligence*, vol. 99, no. 1, pp. 21–71, 1998.
- [3] K. Konolige, E. Marder-Eppstein, and B. Marthi, “Navigation in hybrid metric-topological maps,” in *ICRA*, 2011.
- [4] B. Kuipers, “The spatial semantic hierarchy,” *Artificial Intelligence*, vol. 119, pp. 191–233, 2000.
- [5] S. Ekvall, D. Kragic, and P. Jensfelt, “Object detection and mapping for service robot tasks,” *Robotica*, vol. 25, no. 2, pp. 175–187, 2007.
- [6] A. Pronobis and P. Jensfelt, “Large-scale semantic mapping and reasoning with heterogeneous modalities,” in *ICRA*, 2012.
- [7] Z. Liu and G. von Wichert, “Extracting semantic indoor maps from occupancy grids,” *RAS*, 2013.
- [8] A. Ranganathan and F. Dellaert, “Semantic modeling of places using objects,” in *RSS*, 2007.
- [9] K. M. Wurm, D. Hennes, D. Holz, R. B. Rusu, C. Stachniss, K. Konolige, and W. Burgard, “Hierarchies of octrees for efficient 3D mapping,” in *IROS*, 2011.
- [10] J. Mason and B. Marthi, “An object-based semantic world model for long-term change detection and semantic querying,” in *IROS*, 2012.
- [11] G. D. Hager and B. Wegbreit, “Scene parsing using a prior world model,” *IJRR*, vol. 30, no. 12, pp. 1477–1507, 2011.
- [12] J. Elfring, S. van den Dries, M. J. G. van de Molengraft, and M. Steinbuch, “Semantic world modeling using probabilistic multiple hypothesis anchoring,” *RAS*, vol. 61, no. 2, pp. 95–105, 2013.
- [13] L. L. S. Wong, L. P. Kaelbling, and T. Lozano-Pérez, “Data association for semantic world modeling from partial views,” in *ISRR*, 2013.
- [14] A. Doucet, J. F. G. de Freitas, and N. J. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*. Springer, 2001.
- [15] “Robust Monte Carlo localization for mobile robots,” *Artificial Intelligence*, vol. 128, no. 12, pp. 99–141, 2001.
- [16] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2005.
- [17] L. L. S. Wong, L. P. Kaelbling, and T. Lozano-Pérez, “Collision-free state estimation,” in *ICRA*, 2012.