This manuscript has been accepted at IROS 2014. Its re-submission to the ICRA 2014 workshop on "Robots in Homes and Industry" was explicitly encouraged by the workshop organizers and has been approved by the IROS 2014 program chairs.

Prior-Assisted Propagation of Spatial Information for Object Search

Malte Lorbach[1] Sebastian Höfer[1] Oliver Brock[1]

***Abstract*— We propose a novel method for efficient object search in realistic environments. We formalize object search as a probabilistic inference problem over possible object locations. The method makes two contributions. First, we identify five priors, each capturing structure inherent to the physical world that is relevant to the search problem. Second, we propose a formalization of the object search problem that leverages these priors effectively. Our formalization in form of a probabilistic graphical model is capable of combining the various sources of information into a consistent probability distribution over object locations. The formalization allows us to sharpen the distribution by determining and propagating the effects of knowledge about the world. We use this reasoning method to select actions of a searching robot in a simulated environment and show that it results in efficient object search.**

I. Introduction

Robotics research aims at enabling robots to perform tasks in real world environments. Common tasks in these environments, such as pick and place or manipulation, require the robot to locate objects, given only partial and uncertain information about the world. This object search may require costly, physical motion. It therefore is desirable that the robot is able to reason about the most probable location of a target object, so as to visit this location first. We define the *object search problem* as the efficient exploration of an environment to find an object. Exploration is performed by first identifying a probability distribution over object locations by leveraging all available knowledge. Based on this distribution, the robot chooses an action, leading to new observations, triggering the computation of a new distribution over locations, etc.

In this paper, we present a novel method for reasoning about object locations in realistic environments such as the apartment in Figure 1. The contribution of our method is twofold. First, we identify five priors, each capturing structure inherent to the physical world. Second, we propose a formalization of the object search problem that leverages these priors effectively. Our formalization is capable of combining the various sources of information into a consistent probability distribution. The formalization allows us to sharpen the distribution by determining and propagating the effects of combining knowledge about the world. We use this reasoning method to select actions of a searching robot in a simulated environment and show that it results in efficient object search.

We gratefully acknowledge the funding provided by the Alexander von Humboldt foundation and the Federal Ministry of Education and Research (BMBF) and through the First-MM project (European Commission,FP7-ICT-248258).

[1]Malte Lorbach, Sebastian Höfer and Oliver Brock are with the Robotics and Biology Laboratory, Technische Universität Berlin, Germany

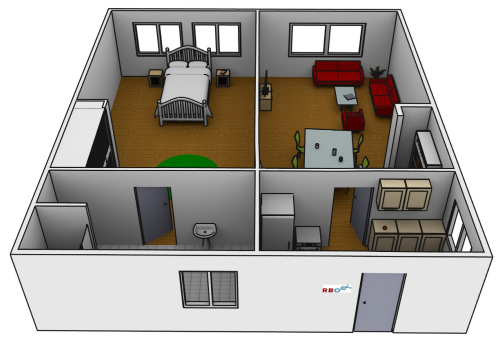


Fig. 1. Searching for objects or places in human-made environments requires effective spatial reasoning. We present a spatial reasoning method that leverages structure inherent to the environment to propagate information so as to improve search performance.

The five proposed priors represent the following information about the structure of the world and the specific scene (the colors identify the priors throughout the paper):

- **Scene Structure (SS)** comprises all facts about the current scene collected by observing objects and their locations.
- **Domain Knowledge (DK)** captures spatial co-occurrence of object pairs. At this point it is important to note that locations are simply treated as objects (locations can contain objects, just as objects can contain objects, and locations may contain other locations, hence the two concepts merge).
- **Physical Constraints (PC)** encode physical facts that constrain the possible locations of objects. For example, large objects cannot be inside smaller objects.
- **Logical Consistency (LC)** further constrains the possible locations of objects using logic. We define two types of consistency: *mutual exclusion* and *transitive consistency*. Mutual exclusion ensures that an object only appears at one place at a time. Transitive consistency enforces the transitivity property of spatial relations: if the laptop is on the desk, and the desk is in the office, then the laptop must also be in the office.
- **Search History (SH)** can be interpreted as short-term memory. The robot remembers the relationships between objects it already has observed and can use this knowledge to guide its search.

To solve the object search problem, we encode these priors in a probabilistic model. By performing inference consistent with these priors, we compute highly informed, globally consistent hypotheses about object locations based on partial and uncertain knowledge of the world.

II. Related Work

To reason about object locations, the robot requires *domain knowledge* that expresses the likelihood of objects occurring at particular locations [1]. Such domain knowledge can be encoded in terms of spatial co-occurrence statistics and then be used to classify locations [2] or guide object search [3]. The concept of spatial co-occurrence for search has been applied to a specific scenario, in which cupboards were the considered locations [4]. In this work, the authors improve reasoning by incorporating *physical constraints* implied by object size, albeit in a narrow application domain.

Another line of research performs efficient object search in large scale environments [5]. The proposed system first selects a limited set of potential locations (rooms) using semantic information (e.g., "subway-shop is of type restaurant-space") and then infers the most likely location within this limited set using spatial co-occurrence. Although the authors use *scene structure* (semantic information) and *domain knowledge* (spatial co-occurrence), the reasoning ends at a very coarse level of abstraction (rooms).

The object search problem can also be considered in unknown environments [6]. *Domain knowledge* in form of spatial co-occurrence can then be used to find a room with a high probability of containing the target. Within that room, visual search is performed [7], assuming knowledge of the exact location of the target in the scene graph (e.g., book in box on table). This work is complementary to ours, as we generate the relational location hypotheses that could serve as input to such a visual search algorithm.

The object search problem was also formulated in the context of supermarkets. Here, *domain knowledge* consists of co-occurrence statistics from typical placement of products [8]. The method also reasons about object attributes (e.g., edible). The likelihood that a shelf contains a particular item depends on products in the proximity (e.g., adjacent shelf) and their attributes. The consideration of related locations can be considered as a form of "short-range" information propagation. Our proposed method, in contrast, propagates information throughout the entire scene graph. However, our method does not consider object attributes.

From this review of related work it becomes apparent that *domain knowledge* is the most crucial ingredient to efficient object search. Moreover, we have seen that *physical constraint* priors (such as size) can help to make search even more efficient. However, the idea of improving object search by using *scene structure* to *propagate* spatial information among locations has not yet been considered explicitly.

III. Probabilistic Inference of Location Hypotheses

We now describe the relevant aspects of the search problem in an incremental fashion. Along the way, we will expand our formalization and address each aspect with an additional prior. The result is a probabilistic formalization of the problem in form of a factor graph. This factor graph allows us to reason about spatial locations and to infer a probability distribution over locations of the target object.

A. Object Search In Real-World Environments

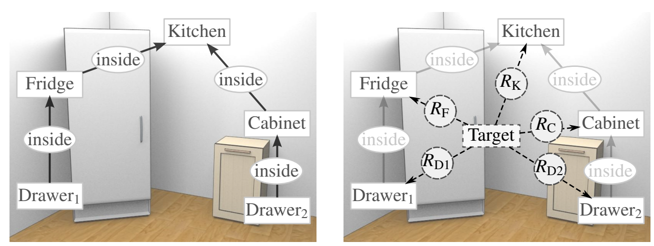
We want to determine a probability distribution over object locations. We assume that the robot possesses the perceptual skills to perceive a scene as a hierarchical structure, i.e. as a *scene graph*, as the one shown in Figure 2(a), containing objects $o_1, \dots, o_n \in \mathcal{O}$ (rectangular nodes) and their *spatial relations* (edges). We consider three types of spatial relations (edge labels): *in*, *on* and *disjoint*, where the latter is equivalent to the lack of an edge in the scene graph. Moreover, we consider *in* and *on* to be transitive.

Given this scene graph, the robot's task is to infer the most likely spatial relations of a target object to all other objects in the scene (the dashed edges in Fig. 2(b)). Formally, we denote the (unknown) spatial relation of the target object s with an object o as a predicate $R(s,o)$ or as R_o when talking about a fixed target object.

To be able to infer the most likely relations, we represent them as discrete random variables and specify a probability mass function P over these variables. The goal is then to find a set of spatial relations that maximizes the posterior over P given the aforementioned priors $\cdot$,

$$\underset{r_{o_1},\dots,r_{o_n} \in \{in, on, disjoint\}}{\arg\max} P(R_{o_1} = r_{o_1}, \dots, R_{o_n} = r_{o_n} | \cdot). \quad (1)$$

We thus compute the instantiation of all relations R_{o_i} between the target object s and all other objects that has the maximum probability, given priors that have yet to be specified. In the forthcoming sections, we are going to formalize P, and explain how to add suitable priors. Each prior captures a different aspect of the object search problem and will ultimately allow us to factorize P and solve Equation (1) efficiently.



(a) Scene graph as observed by the robot

(b) We reason about the target's unknown spatial relations R_{o_i}

Fig. 2.

B. Representing Probabilistic Location Hypotheses

To render the computation of the probability distribution P efficient, or even tractable, we leverage the priors mentioned above. We now explain how to incorporate the first three priors: *domain knowledge (DK)*, *physical constraints (PC)*, and *search history (SH)*.

Each of these priors is represented by a discrete probability distribution for every spatial relation R_{o_i}, as depicted in Figure 3. For now, it suffices to introduce these three priors. In Section III-D, we will explain how to combine them to a joint distribution for the spatial relation. Given the priors, we rewrite Equation (1):

$$\operatorname*{arg\,max}_{r_{o_1},\dots,r_{o_n}\in\{in,on,disjoint\}} \quad P(R_{o_1}=r_{o_1},\dots,R_{o_n}=r_{o_n} |\ \mathrm{DK},\mathrm{PC},\mathrm{SH}), \tag{2}$$

by adding the concrete priors. This formalization allows us to infer the most likely location hypothesis, but this hypothesis will not necessarily be consistent with the scene structure.

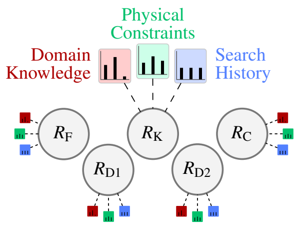


Fig. 3. Every random variable represents a spatial relation and captures domain knowledge, physical constraints and search history.

C. Inferring Consistent Location Hypotheses

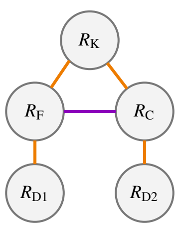


Fig. 4. Connecting random variables enables propagation of spatial information.

To enforce scene consistency in our location hypotheses, we leverage the *scene structure (SS)* prior. The scene structure is captured by the edges in the scene graph (Fig. 2(a)). We incorporate the scene structure into our probabilistic representation by connecting a pair of random variables R_{o_i} and R_{o_j} iff o_i and o_j are connected in the scene graph. The resulting edges in Figure 4 are colored orange. This step turns our probabilistic representation into a graphical model [9], which now allows information to be propagated among random variables.

Although we can now propagate information according to the scene structure, propagation can still result in a globally inconsistent state. The reason is that the true spatial relations $R_{o_i} = r_{o_i}$ are yet unknown, and not every combination is logically possible. Hence, we prevent information propagation of inconsistent states by incorporating *logical consistency (LC)* into the graph. We do so by restricting information flow along the already created (orange) edges to reflect transitivity consistency. Moreover, we add further connections between the random variables (purple edges in Fig. 4) to deal with mutual exclusion.

D. Factor Graph Representation

The graphical model in Figure 4 captures all aspects necessary for probabilistic reasoning about object locations. However, the model is impractical from a computational perspective due to the presence of loops. The model also makes it difficult to combine different probability distributions based on the priors (probability masses in Section III-B) and to enforce global consistency with other priors (Section III-C). Therefore, in this last step, we transform the graphical model into a factor graph.

A factor graph is a bipartite graph which connects random variables by potential functions. Each factor is a function that maps the states *Val* of the connected variable(s) to a non-negative potential:

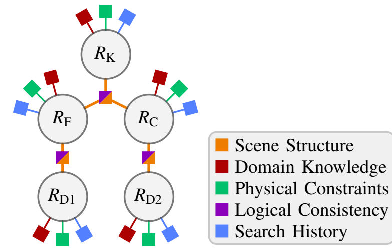


Fig. 5. The factor graph efficiently implements the five priors that leverage our knowledge.

$$\phi(\cdot) : \mathit{Val}(\cdot) \mapsto \mathbb{R}^+. \tag{3}$$

The potential of a factor can be determined by combining several probability distributions. To achieve this, we chose to represent the factors as Boltzmann distributions, which relate energies to potentials. That is, we define the factor functions to be of the form:

$$\phi(\cdot) := \exp\left(-E(\cdot)\right) \tag{4}$$

in which E is an energy function that in turn consists of several energy terms.

In order to transform the graphical model into a factor graph, we *1)* turn the probability mass functions from our domain knowledge, physical constraints, and search history priors from Section III-B into univariate factors and *2)* replace the dependencies between the variables in the graphical model from Section III-C by suitable multivariate factors. We will now describe these two types of factors.

1) Univariate factors for domain knowledge, physical consistency and search history: We first define three factors ϕ_{DK}, ϕ_{PC} and ϕ_{SH} for the corresponding priors. We connect to every random variable R_{o_i} a factor of each of these three types. The colored squares at each random variable in Figure 5 illustrate this. In our implementation we merge the three factors into one common factor $\phi_{\mathrm{DK,PC,SH}}$ to gain computational efficiency. The combined factor's energy is the weighted sum of energy terms of each single factor:

$$\mathrm{E}_{\mathrm{DK,PC,SH}}(R_o) = \lambda_{\mathrm{DK}}\mathrm{E}_{\mathrm{PC}}(R_o) + \lambda_{\mathrm{PC}}\mathrm{E}_{\mathrm{PC}}(R_o) + \lambda_{\mathrm{SH}}\mathrm{E}_{\mathrm{SH}}(R_o) \tag{5}$$

with weights λ_{DK}, λ_{PC}, and λ_{SH}.

2) Multivariate factors for scene structure, physical, and logical consistency: To maintain the ability to propagate information among random variables, we first identify cliques in the graphical model. For every clique $\mathcal{R} \subseteq \{R_{o_1},\dots,R_{o_n}\}$, we create a new factor $\phi_{\mathrm{SK,LC}}$ and connect all of the clique's members to this factor. This type of factor has orange/violet color in Figure 5, symbolizing that it originates from our scene structure and logical consistency priors. Given the clique $\mathcal{R}$, we compute the energy for $\phi_{\mathrm{SK,LC}}$ as the weighted sum of two energy terms

$$E_{\mathrm{SK,LC}}(\mathcal{R}) = \lambda_{\mathrm{ME}}\mathrm{E}_{\mathrm{ME}}(\mathcal{R}) + \lambda_{\mathrm{TC}}\mathrm{E}_{\mathrm{TC}}(\mathcal{R}), \tag{6}$$

capturing the two logical consistency rules, namely mutual exclusion and transitivity consistency, respectively.

Our factor graph now allows us to express the posterior from Equation (1) as the product of all factors ϕ_i,

$$P(R_{o_1} = r_{o_1}, \ldots, R_{o_n} = r_{o_n} | \mathrm{DK}, \mathrm{PC}, \mathrm{SH}, \mathrm{SK}, \mathrm{LC}) = \frac{1}{Z} \prod_i \phi_i, \tag{7}$$

where Z is a normalization factor. Using this factorization, we can efficiently compute the solution to our problem stated in Equation (1) by applying Belief Propagation [10]. The result of the Belief Propagation algorithm constitutes our final location hypothesis for the target object which is consistent with all of the priors introduced above.

IV. Experiments

We evaluate the performance of our method in experiments by searching for objects in a simulated world. We measure performance in terms of the expected number of actions required to find an object divided by the expected optimal number of actions. Thus, optimal performance is 1, all other numbers can be interpreted as a factor expressing the sub-optimality of the performance. All results reported are averaged over 30 independent trials in the same environment.

A. Experimental Setup

1) Simulated World: We conduct the experiments in a simulated apartment (see Fig. 6) consisting of five rooms containing 35 pieces of furniture (cupboards, tables, a sofa etc., note that a cupboard with two shelves counts as three objects) and 51 items (cups, soap, socks, etc.). Rooms, furniture and items are all considered to be objects of which some have an open/closed state (e.g., rooms and cupboards).

The robots can "see" the scene graph of the room it currently is in. However, if an object is in the closed state, it cannot perceive the content of that object. The robot is able to perform "open" actions on objects that are closed and have been perceived by the robot. At the beginning of every experiment, all objects are closed.

In the absence of initial scene graph knowledge, the robot only knows one closed object: the apartment itself. Its first action will be to open the door to the apartment, then being able to perceive the hallway with closed doors to the other rooms. For each experiment, we count the number of "open" actions the robot has to perform to find the target object.

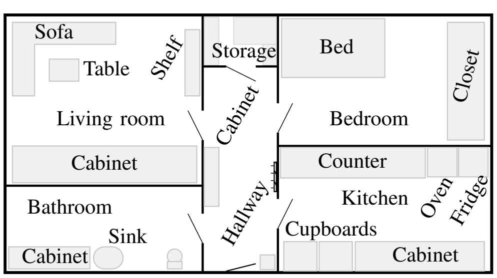


Fig. 6. Map of the experimental environment (objects not shown)

2) Inference Strategies: We compare our *propagation approach* to a *greedy approach*. The greedy approach exploits domain knowledge, physical constraints, and search history priors, but not the scene structure and logical consistency priors, and therefore resembles the search approaches proposed in related work [3]–[8]. We implement the greedy approach by setting the multivariate factor weights $\lambda_{\mathrm{TC}}, \lambda_{\mathrm{LC}}$ to zero, which is equivalent to removing all edges from the factor graph (see Sec. III-D.2). Otherwise we set $\lambda_{\mathrm{DK}} = \lambda_{\mathrm{PC}} = 3$, $\lambda_{\mathrm{SH}} = 1$, $\lambda_{\mathrm{ME}} = \lambda_{\mathrm{TC}} = 25$. By comparing the two methods, we can examine the effect of propagation on search efficiency.

We also want to examine the influence of a priori knowledge of the scene structure on the performance. We therefore compare the situation in which the robot starts an experiment with no prior knowledge of the scene to a situation in which the robot starts with a map of the apartment. The map is a scene graph containing the six rooms and the closed furniture inside them but not the "invisible" locations inside the closed furniture or the objects.

3) Search Policy: We embed each of the four reasoning strategies into the same observation-reasoning-action loop. The robot first observes a scene graph. From this scene graph, a factor graph is constructed and used to infer a probability distribution for the location of the target object. To test the location hypothesis, the robot identifies which room/object it needs to open and executes the respective action. Following its action, the robot makes a new observation. If this observation includes the target object, the search ends successfully. Otherwise the reasoning mechanism updates the search history factors with the new observations and the process repeats.

4) Gathering Spatial Co-occurrence Information: Our approach to object search leverages information from spatial co-occurrence statistics of objects. In some of our experiments, these statistics are hand-coded. But we also evaluate our method with real-world spatial co-occurrence data, which was obtained as follows. Similar to prior work [11], we use an image search engine[1] to extract spatial co-occurrence data from the WWW. We estimate the spatial co-occurrence for the two relations *in* and *on* for object/location pairs based on search statistics. For example, we count the number of results for a query like "knife in the kitchen" or "cup on the counter" and normalize by the counts of "in the kitchen" or "on the counter". The count for the *disjoint* relation is computed by assuming a maximum total number of occurrences of each object in the database and subtracting the count of the *in*-query and *on*-query from it.

5) Framework: The reasoning system uses an inference algorithm (Belief Propagation) that is provided by the libDAI library [12]. A typical iteration of the algorithm (a full approximation of the probability mass and extracting the MAP state) takes less than one second on an eight-core PC with 2.3GHz each.

[1]http://www.bing.com/images

B. Evaluation Using Hand-Coded Domain Knowledge

To analyze the benefits of our reasoning approach, we construct three scenarios suitable for illustrating the propagation of information. To eliminate the effects of noisy domain knowledge, we manually define the energies that represent the spatial co-occurrence information (Eq. 5).

1) Scenarios: The three scenarios show how propagation of scene and partial domain knowledge allows the robot to infer the correct location hypothesis. The first two cases illustrate that information spread across the scene can be leveraged by propagation. We consider two targets: "ice cream" and "book". We provide the domain knowledge that ice cream is usually found in kitchens, in freezers and slightly less often in storages. In the experiment apartment, the ice cream is located "in freezer in storage". Similarly, we provide that books are equally likely to be in living rooms, bedrooms and kitchens, and often in shelves. The true location of the book is "in shelf in living room". In both cases propagating the information about freezer/shelf to the corresponding room, helps to find the target quicker.

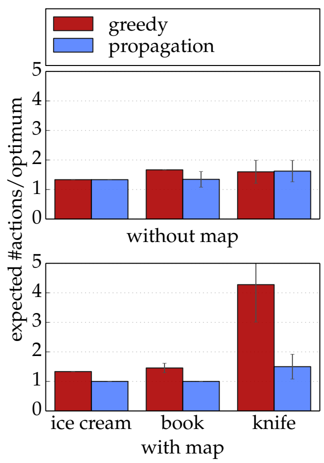


Fig. 7. Object search performance with hand-coded domain knowledge: expected number of actions required to find the object, relative to the optimal solution (performance of 1 is optimal, performance of 2 means twice as many actions required as in the optimal solution)

In the third scenario we point out a search problem in which a huge search space is reduced by propagating domain knowledge. We choose "knife" as the target object and provide the domain knowledge that knifes usually occur in drawers as well as in kitchens. Without combining these two pieces of information all drawers in the apartment are equally likely and, therefore, span a huge search space. Our strategy propagates the information and should, thus, correctly infer "knife in drawer in kitchen" as the most probable object location.

2) Results and discussion: The results for the first experiment show that propagation successfully leverages information where it is available (Fig. 7). Without map, neither strategy reaches optimal performance. This is because the information available initially is not sufficient to reduce the search space far enough. In all cases several location alternatives remain (several equally likely rooms or drawers) all of which need to be checked. Without map, the problems cannot be simplified.

However, when the robot does have initial knowledge about the environment (with map), the propagation approach requires fewer actions than greedy in all scenarios. This result confirms our expectation: the ice cream and the book are found with fewer actions (decrease by approx. 50%) by our method because we propagate information from the freezer and the shelf, respectively, "bottom-up" and, thus, guide the robot immediately to the right rooms. Note that the difference between the strategies would be even bigger if the scene had more alternative locations to check. This becomes clear in the knife scenario in which the number of drawers is huge and only propagating information from the kitchen "top-down" restricts this search space. Propagation decreases the number of actions for finding the knife by a factor of approx. four.

The analysis of the search problems illustrate the strength of our reasoning method: propagating information improves search efficiency in suitable scenarios.

C. Evaluation on Real-World Domain Knowledge

We now investigate the performance of our method with noisy domain knowledge (see Section IV-A.4). We gather spatial co-occurrence data for 46 typical household objects (Fig. 8) and evaluate our method by searching for 15 objects.

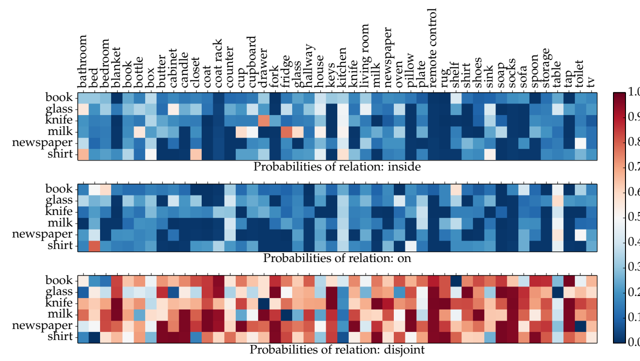


Fig. 8. A subset of the spatial co-occurrence probabilities used in our experiments regarding the relations *in*, *on* and *disjoint* between the objects on the y-axis and the objects on the x-axis

1) Results: The results for the 15 target objects are shown in Figure 9. They confirm the contribution of propagation to location reasoning. Without map the propagation strategy dominates the greedy strategy except when searching for the glass, the shirt and the shoes. With map propagation find everything apart from the shoes quicker than greedy. The propagation strategy performs optimally for three objects (remote control, book and candle) and sub-optimally for the other objects.

2) Discussion: Analyzing the results from the real-world experiment, we identify three prototypical problem cases which lead to poor performance:

a) Indistinguishable objects: Similar to the experiment with hand-coded domain knowledge, the robot has no means of distinguishing different objects of the same type in one room. This is the case for knife (in one of four kitchen drawers) or plate (in one of two kitchen cupboards). This problem affects both strategies equally and may be solved by including object attributes to distinguish locations of the same type [8].

b) Ambiguous concepts: Ambiguous terms in language lead to false co-occurrence information. "Glass", for example, is ambiguous as it can be associated with a drinking container, windows or glass tiles on a wall. Consequently,

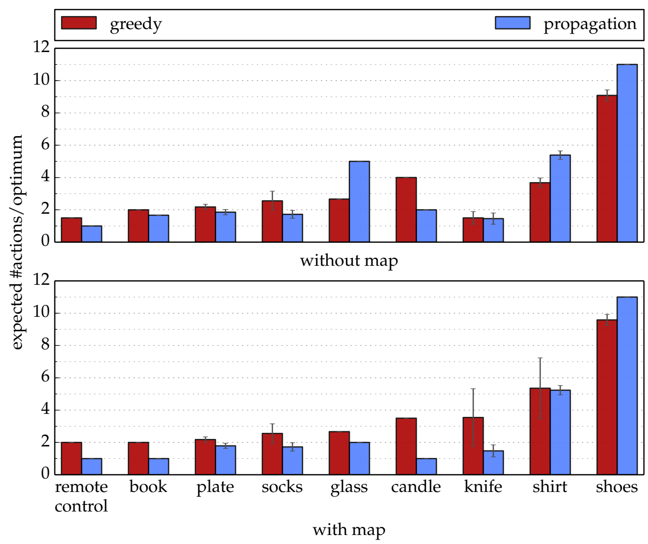


Fig. 9. Object search performance with automatically generated, noisy domain knowledge: expected number of actions required to find the object, relative to the optimal solution (performance of 1 is optimal)

the query to the image search engine returns hits for all of those meanings and, thus, the co-occurrence probabilities are skewed (e.g., consider the high probability for "glass in bathroom" in Fig. 8). We could solve this problem by incorporating better sources for domain knowledge, such as common sense databases (e.g., ConceptNet: `http://conceptnet5.media.mit.edu`).

c) Misleading knowledge: In some cases, the spatial co-occurrence knowledge contradicts the setup of our particular scene. For example, according to our domain knowledge, the most probable location of shoes is on the bed, although they are located in the cabinet in the hallway. The same occurs for the shirt, which we placed in a drawer in the bedroom closet, but the domain knowledge identifies locations in the bathroom and the kitchen as highly probable. Obviously, the domain knowledge extracted from image tags lacks common sense knowledge, as people tend to describe noteworthy things rather than ordinary and mundane relations. Other authors made similar observations [3], [11].

V. Conclusion

We presented a novel method to generate globally consistent location hypotheses for object search. Our contribution is a probabilistic model that is able to propagate spatial information among dependent locations by exploiting the consistency of the scene. We define five priors to assist this propagation by constraining and informing information flow. These priors capture structure inherent to the physical world and relevant to the search problem: knowledge about the structure of the specific arrangement of rooms and objects in the environment, statistical information about co-occurrence of objects, constraints on object locations based on physics, constraints on beliefs about object locations that must be satisfied in a physical world, and the robot's own observations. We present experiments in a simulated environment to demonstrate that our method leverages the information available in the priors and from observations of the scene to improve the efficiency of object search, even when using noisy and partially incorrect domain knowledge extracted via web search.

References

[1] M. J. Schuster, D. Jain, M. Tenorth, and M. Beetz, "Learning organizational principles in human environments," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*. IEEE, May 2012, pp. 3867–3874.

[2] P. Viswanathan, D. Meger, T. Southey, J. J. Little, and A. K. Mackworth, "Automated Spatial-Semantic Modeling with Applications to Place Labeling and Informed Search," *Canadian Conf. on Computer and Robot Vision*, pp. 284–291, May 2009.

[3] T. Kollar and N. Roy, "Utilizing object-object and object-scene context when planning to find things," in *Proc. of the IEEE Int. Conf, on Robotics and Automation (ICRA)*, May 2009, pp. 2168–2173.

[4] L. L. S. Wong and L. P. Kaelbling, "Manipulation-based Active Search for Occluded Objects," in *Proc. of the IEEE Int. Conf, on Robotics and Automation (ICRA)*, 2013.

[5] L. Kunze, M. Beetz, M. Saito, H. Azuma, K. Okada, and M. Inaba, "Searching objects in large-scale indoor environments: A decision-theoretic approach," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, May 2012, pp. 4385–4390.

[6] A. Aydemir, A. Pronobis, M. Gobelbecker, and P. Jensfelt, "Active Visual Object Search in Unknown Environments Using Uncertain Semantics," *IEEE Transactions on Robotics*, vol. 29, no. 4, pp. 986–1002, Aug. 2013.

[7] K. Sjöö, A. Aydemir, and P. Jensfelt, "Topological spatial relations for active visual search," *Robotics and Autonomous Systems*, vol. 60, no. 9, pp. 1093–1107, Sept. 2012.

[8] D. Joho and W. Burgard, "Searching for objects: Combining multiple cues to object locations using a maximum entropy model," in *Proc. of the IEEE Int. Conf, on Robotics and Automation (ICRA)*, May 2010, pp. 723–728.

[9] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[10] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[11] M. Hanheide, C. Gretton, and R. Dearden, "Exploiting probabilistic knowledge under uncertain sensing for efficient robot behaviour," *Proc. of the Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 2011.

[12] J. M. Mooij, "libDAI: A Free and Open Source C++ Library for Discrete Approximate Inference in Graphical Models," *Journal of Machine Learning Research*, vol. 11, pp. 2169–2173, Aug. 2010.