Object search on a mobile robot using relational spatial information

Alper AYDEMIR, [1] Kristoffer SJÖÖ and Patric JENSFELT
Centre for Autonomous Systems, Royal Institute of Technology, Sweden

Abstract. We present a method for utilising knowledge of qualitative spatial relations between objects in order to facilitate efficient visual search for those objects. A computational model for the relation is used to sample a probability distribution that guides the selection of camera views. Specifically we examine the spatial relation "on", in the sense of physical support, and show its usefulness in search experiments on a real robot. We also experimentally compare different search strategies and verify the efficiency of so-called indirect search.

Keywords. Indirect search, Active visual search, Spatial relations, Qualitative spatial reasoning

Introduction

The ability to find objects in a 3D world is an important item on a mobile robot's skill repertoire. Previous work on object search stems mainly from the field of computer vision. Ideally a robot with a specific task of locating an object should make use of all the bits and pieces of evidence; be it from an overheard dialogue, target object's class limiting the search to a specific region (e.g. forks are usually found in kitchen) or a known spatial relation between the target and some other entity. Some work concentrates on locating the target in the image, thus assuming that the target is already in the field of view [7]. Others investigate algorithms for covering a known or previously unknown world efficiently [1,5,8,9,10].

One powerful idea which naturally involves integration of multiple cues is *indirect search* [3]. Indirect search is about first looking for an intermediate object in order to find the target object by exploiting the relation between the former and the latter. This can be exemplified by first searching for the larger and easier-to-detect whiteboard, and then looking for the pen next to it. To be practical, the system needs to make a decision on which approach to choose based on some criteria. Although this is a simple idea, accomplishing it by fusing multiple types of cues can prove to be hard and is not yet in place in the previous work.

The novelty of this paper is given by an investigation of the following question: Is it possible to make use of spatial relations in order to aid a mobile robot tasked with finding an object? For this particular work we have chosen to investigate the relation of

[1]Corresponding Author: Alper Aydemir, Royal Institute of Technology (KTH), Centre for Autonomous Systems, SE-100 44 Stockholm, Sweden; E-mail: aydemir@csc.kth.se

physical support, i.e. *on*. We introduce a computational perceptual model for the physical support relation, and show how algorithms using this model can significantly increase the efficiency of visual object search, illustrating the fact through real world experiments. In this way, we believe that the work presented here takes a more principled approach towards indirect search compared to previous work.

1. Spatial Relations as functions

Spatial relations between entities are important in human cognition, as evidenced by the prolific use of spatial prepositions in language, in both concrete and metaphorical contexts. Here, we are interested in using the information carried by a relation between two objects A and B, together with the location of one of them, for the purpose of locating the other efficiently.

We regard a spatial relation as a function, dependent on the objects involved, from the space of all the objects' possible poses, to the interval $[0, 1]$:

$$\mathcal{R}_{A,B} : \{\pi_A, \pi_B\} \rightarrow [0, 1] \tag{1}$$

where 1 represents that the relation is completely fulfilled by the pose combination, and 0 that the relation does not apply at all. The resulting value, despite being in the range $[0, 1]$, is not a probability. However, it is possible to obtain a probability distribution over poses implicitly from this function, as shown below.

1.1. ON

As a good example of a spatial relation that will be useful for a robot in a search scenario, we have chosen "on". "On" is one of the most fundamental prepositions in the English language, and represents a highly relevant functional relationship between many objects in our environment [4,6]; thus, a robot will often have information about an object's location in terms of it being "on" something else – this information could come from dialogue with humans, from commonsense rules for the typical behaviour of objects, or from a statistical model learned from experience over time.

The central functional aspect of the word "on" is the *support* that one object gives another. Humans learn to judge this with experience, manipulating and observing; for now, robots must rely on short-cuts. We therefore propose a perceptual geometric model intended to estimate how well the relation between two objects corresponds to one of support. The model is defined using the following criteria (O denotes the *trajector* object, i.e. the object that is "on" the other, and S the support object or *landmark*). The proposed function is termed $\textsc{On}(O, S)$. The criteria are illustrated in Figure 1; they are:

- *Separation between objects*, d. d can be positive or negative, negative values meaning that objects are, or seem to be, interpenetrating.
  In order for an object to mechanically support another, they must be in contact. Due to imperfect visual input and other errors, however, contact may be difficult to ascertain precisely. Hence, to create a soft constraint, the apparent separation is used as a penalty.

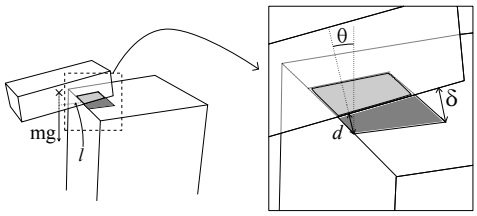


Figure 1. Key features used in computation of ON: Separation d, COM offset l, contact angle θ and contact threshold δ. The gray area represents the contact.

- *Horizontal distance between COM and contact*, l. It is well known that a body O is statically stable if its center of mass (COM) is above its area of contact with another object S; the latter object can then take up the full weight of the former. Thus we impose a penalty on $\text{ON}(O, S)$ that increases with the horizontal distance from the contact area to the COM of O. The contact area is taken to be that portion of S's surface that is within a threshold, δ, of O, in order to deal with the uncertainties described above. If $d > \delta$, the point on S closest to O is used instead; otherwise, l is the positive distance to the outer edge of the contact area if outside it, and the negative distance if inside.
- *Inclination of normal force*, θ – the angle between the normal of the contact between O and S on the one hand, and the vertical axis on the other. The reason for including this is that, all other things being equal, the normal force decreases as the cosine of θ, meaning the weight of O must be either supported by another object or by friction (or adhesion).

All these values can be computed from visual perception in principle. The position of the COM is taken as the average point of the objects' geometry (since density cannot be determined by vision), unless otherwise known in advance.

The first criterion is evaluated as the *distance factor* in an exponential function:

$$\text{ON}_{distance}(O, S) \triangleq \exp\left(-\frac{d}{d_0(d)} \ln 2\right) \tag{2}$$

where d_0 is the falloff distance at which ON drops by half:

$$d_0 = \begin{cases} -d_0^-, d < 0 \\ d_0^+, d >= 0 \end{cases}$$

The constants d_0^- and d_0^+ are both greater than 0 and can have different values (representing the penalty for the penetrating and nonpenetrating case, respecively).

The latter two criteria make up the *contact factor*:

$$\text{ON}_{contact}(O, S) \triangleq \sin\theta \cdot \frac{1 + \exp(-(1-b))}{1 + \exp\left(-\left(\frac{-l}{l_{\max}} - b\right)\right)} \tag{3}$$

Here, $l_{\max}$ is the maximum possible distance an internal point can have within the contact area, and b is an offset parameter.

The exact expressions for the factors (2) and (3) are not central here; what matters is that they yield the applicability 1 for the ideal case for each criterion, and drop off to 0 as the criterion is violated, while being "soft" in order to be robust to error.

The values are combined by choosing whichever factor is smaller, indicating the greater violation of the conditions for support:

$$\textsc{On}(O, S) \triangleq \min(\textsc{On}_{contact}, \textsc{On}_{distance}) \tag{4}$$

1.2. Probability modelling

Although the conceptualization above does not explicitly make use of any probabilities, it is obvious that the fact of an object being ON another is not sufficient to recover the exact pose of the trajector. A probability distribution over poses can be produced in the following way:

Given the pose and geometry of the landmark S, and the geometry (but not the pose) of the trajector O, each possible pose π for the trajector yields a value of $\textsc{On}(O_\pi, S)$ for that pose.

It is now possible to introduce probabilities in the following way. Introduce a true/false event $On(O, S)$ signifying that $\textsc{On}(O, S) > t$ where t is a threshold. Then,

$$p(\pi|On(O_\pi, S)) = \frac{p(On(O_\pi, S)|\pi)p(\pi)}{p(On(O_\pi, S))} = \tag{5}$$
$$= \frac{[\textsc{On}(O_\pi, S) > t]p(\pi)}{p(On(O_\pi, S))}$$

Here [] denotes the Iverson bracket:

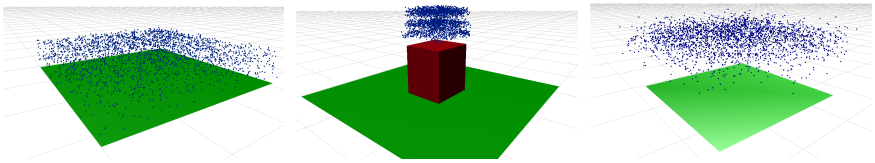
$$[X] = \begin{cases} 1, \text{ if } X \text{ is } \textsc{True} \\ 0, \text{ otherwise} \end{cases}$$

In other words, the probability is simply proportional to the prior for the pose π whenever $\textsc{On}(O_\pi, S) > t$, and 0 elsewhere. Though it may be hard to express this distribution analytically, by drawing samples randomly from $p(\pi)$, discarding those failing to reach the threshold, and normalising over the remainder, an arbitrarily good approximation can be found. In the following, we use a t value of 0.5.

Figure 2 shows simulated examples of distributions sampled according to the above. 2(c) shows *chained* sampling: an object is ON another, which is ON the table, but both have unknown poses. First the bottom object is sampled, and for each sample that passes the threshold, the top object is sampled in turn. The uncertainties of both objects add up, resulting in a more diffuse point cloud at a greater height above the table.

2. Object Search

The goal of the object search process performed by a mobile robot is to calculate a set of sensing actions with minimum cost which brings the target object, in whole or partly, into the sensor field of view so as to maximize the target object detection probability.



(a) Object on table (b) Object on box (c) Object on box on table

Figure 2. Simulated examples of sampled distributions of ON

Here we briefly give a formulation of the object search problem using the notation of [10]. Let Ψ be the $2D$ search region whose structure is known *a priori*. To discretize the search region, Ψ is tessellated into identically sized cells, $c_1 ... c_n$. The area outside of the search region is represented by a single cell c_0. A sensing action s is then defined as taking an image of Ψ from a view point v and running a recognition algorithm to determine whether the target object o is present or not. In the general case, the parameter set of s consists of camera position (x_c, y_c, z_c), pan-tilt angles (p, t), focal length f and a recognition algorithm a; $s = s(x_c, y_c, z_c, p, t, a)$. The cost of a search plan $S = s_0 ... s_i$ is then given as $C(S)$.

A search agent starts with an initial probability distribution (PDF) on target object location over Ψ. We assume that there is exactly one target object in the environment either inside or outside the search region. This means that all cells will be dependent and every sensing action will influence the values of all cells. Let β be a successful detection event and α_i the event that the center of o is at c_i. The probability update rule after each s with a non-detection result is then:

$$\mathbf{p}(\alpha_i|\neg\beta) = \frac{\mathbf{p}(\alpha_i)(1 - \mathbf{p}(\beta|\alpha_i))}{\mathbf{p}(\alpha_0) + \sum_{j=1}^{n} \mathbf{p}(\alpha_j)(1 - \mathbf{p}(\beta|\alpha_j))} \tag{6}$$

Note that for $i = 0$, $\mathbf{p}(\beta|\alpha_i) = 0$, i.e. we cannot make a successful detection if the object is outside the search region . Therefore after each sensing action with a non-detection result the probability mass inside Ψ shifts towards c_0 and the rest of Ψ which was not in field of view.

2.1. Next best view selection

The next step is to define how to select the best next view given a PDF. First, candidate robot positions are generated by randomly picking samples from the traversable portion of Ψ. This results in several candidate robot poses each with associated view cones. For a given camera, the length of the view cone is given by the greatest distance at which the object can reliably be detected, which depends on the size of the object.

The next best view point is then defined as:

$$\underset{j=1..N}{\operatorname{argmax}} \sum_{i=1}^{n} \mathbf{p}(c_i) V(c_i, j) \tag{7}$$

Where N is the number of candidate view points and V is defined as:

$$V = \begin{cases} 1, \text{ if } c_i \text{ is inside of the } j^{\text{th}} \text{ view cone} \\ 0, \text{ otherwise} \end{cases}$$

3. Experiments

3.1. Implementation Details

The robot used in our experiments is a Pioneer III wheeled robot, equipped with a Hokuyo URG laser range finder and a stereo camera (with no zoom capability) mounted on a pan-tilt unit at 1.4 m above the ground. The system uses a SLAM implementation [2] for localization and mapping and builds an occupancy gridmap based on laser data. The experiments were carried out in a mock-up living room (Figure 3). Two planar objects – a low table and a large desk – were present in the experimental area, and their poses known to the system. The detectable objects used were a large cardboard box and small rice carton (see Figure 4). Preparatory experiments showed that the threshold distance, at which the objects were detected at least 75% of the time, was 1 m and 4 m for the small and the large object, respectively. These were the maximum distances used in the view cone generation (see Section 2.1).

Figure 3. Experimental environment and robot platform

During experiments, the larger box and the rice carton were placed randomly on one of the tables, at a 50% chance for each. In order to minimize the bias, different people from our lab, unconnected with the research, were asked to "put the box on the table/desk and rice carton on the box". The objects were free to be placed in any orientation and pose provided they are placed on their physical support object.

In order to assign a prior to the grid cells (Section 2) we generated random samples as described in Section 1.2 and used KDE. 150 samples that passed the threshold $\text{ON} > 0.5$ were convoluted with a simple 2D Epanechnikov kernel:

$$K(u) = c \cdot (1 - u^2)$$

with a kernel radius chosen to be 0.2 m. The resulting grid was then normalized.

Figure 4. Test objects: “rice” and “printer”

The object search was carried out as described in Section 2. The initial information given to the system was:

1. The *a priori* probability that the object sought was in fact in the room was given at 80% (i.e. $\mathbf{p}(c_0) = 0.2$).
2. The “rice” object was ON the “printer” object with 100% certainty.
3. The “printer” object was ON either the table or the desk, each with 50% probability.

When the best next view was decided on, the robot moved to the corresponding position and orientation. 25 pose samples for the target object (with ON above the threshold 0.5) were then obtained from the region of the view cone, and their average used to set the tilt angle of the camera in order to capture the most likely object height.

Object detection and pose estimation was done using previously trained SIFT features. The generation and processing of new views was kept up until either the “rice” object was found, or until the search was considered to have failed. The criterion for failure was a posterior probability of 70% that the object was not inside the room. We performed three types of searches utilising the prior information to varying degrees; uninformed search, chained inference with 2 relations and indirect search with 2 relations. In the following we will denote the rice carton by *A* and the cardboard box by *B*.

3.2. Chained inference with 2 relations

In this test, the information given was that B was ON a table, and that A was ON B, but otherwise A and B had unknown poses. The robot is tasked to look directly for A. By making use of the a priori information via chained inference, as described in Section 1.2, a probability distribution was sampled for A’s pose, and visual search was planned using this distribution directly. Figure 5 shows the robot processing a view during this search. Note the tilt of the camera, illustrating the robot’s expectation for the vertical position of the object, given that it is supposed to be on top of the larger box.

3.3. Indirect search with 2 relations

In this scenario, the robot exploits the position of the relatively more easily detectable B to find A. The initial information provided to the robot is the same as in Section 3.2. However, this time the system first sampled the distribution of B (given that it was on a table) and performed the visual search for B based on the resulting probability distribu-

Figure 5. Chained inference, direct search: While searching for the rice carton, the robot looks towards the height of the target object had it been on top of the large box object.

(a) (b)

Figure 6. (a) The robot first finds the cardboard box which can be detected easily as opposed to rice carton. (b) Once the cardboard box is found, the search space is greatly reduced and the rice carton is found with the next view.

tion. Only if and when B was found did the system compute the distribution of A using this new data, using that distribution in its turn to perform a focused search for A. Note that by finding B and generating possible poses for A the robot reduces the search space significantly. The experimental results also bear this out. Figure 6 shows the robot as it detects the larger box at a distance, then closes to locate the "rice" object at a distance where the model indicates that detection is likely.

3.4. Uninformed search

As a baseline, we ran the algorithm without utilising the information in the spatial relation. Thus, item 3 in the above list of *a priori* knowledge was not used. Instead, the visual search for B used a prior PDF that simply assigned a uniform probability for the object to all obstacles registered by the laser scanner. In lieu of the vertical information otherwise provided by the spatial relation, the camera instead tilted to a set sequence of: down 30°, straight forward, and up 30°. When the B object was detected, the conditioned probabil-

Mode	% success	Avg. # views, failure	Avg. # views, success
Direct chained	73	5	5
Indirect chained	93	5	2
Uniform	46	17	10

Figure 7. Results of experimental evaluation

ity for A was used as in 3.3. The reason for not conducting an uninformed search directly for A over the whole space is that this proved infeasible in experiments, as the number of view points invariably exceeded our limit of 20. The fail search criterion was also not met because the smaller view cones resulting from the object's smaller size shifted little of the probability mass out towards $\mathbf{p}(c_0)$, the probability that the target object is outside of the search space. This is in contrast to the larger "cardboard" object where after each non-detection a substantial amount of probability mass flowed towards $\mathbf{p}(c_0)$.

3.5. Results

For each of 15 different object configuration, all three types of searches were performed for a total of 45 runs. We present the results of our experiments in Table 7.

By comparing uniform and direct search, the advantage of using the spatial relation knowledge is evident. Ignoring the information that the printer box is on the table leads to unnecessary views of the walls and other irrelevant obstacles. Also the lack of vertical position information necessitates redundant image processing as the camera goes through 3 tilt angle settings in order to ensure vertical coverage.

The difference in performance between indirect and direct search illustrates the usefulness of indirect search, even when the spatial relations are taken into account fully. Chained sampling allows the robot to directly create a probability representation for the sought object, bypassing the search for the larger object and providing an approximate height at which to aim the camera; nevertheless, the small size of the object means that many views may be necessary to cover a large area . However in the indirect search case, once the larger object is located then the search space is greatly reduced and typically the target object is found within the next view or two.

4. Conclusions

We have proposed a way in which spatial relations, in the form of applicability functions, can be used to aid in visual object search. We suggested a perceptual geometrical model that approximates the core meaning of the topological preposition "on", i.e. the notion of support. In experiments on real robot, running autonomously, we have shown the advantages to being able to incorporate information about support into a visual search framework:

- Knowing that a relation holds between an object of known pose and one of unknown pose allows for limiting the 2D space over which to search for the latter.
- Indirect search can help with the localization of a smaller object, by allowing the search to start with a larger, easier-to-detect object.
- The support property can be used to guide the search in the vertical dimension.

The results reinforce the notion that indirect search is a useful method in active visual search; our contribution here is the expression of how indirect search is done in conjunction with qualitative spatial relations, as well as the specific instantiation using the ON relation.

5. Discussion

In this work, experiments were relatively limited in scope and served only to compare different search modes with each other. One avenue of investigation is to vary the parameters of the objects involved; for example, changing the characteristics of the involved objects to find the threshold where indirect search becomes more costly than chained search.

The inclusion of other qualitative relations is another interesting direction for further research; especially other topological relations such as “in”, “near”, and “at”, as these are all to some extent objective and functional in nature.

The search problem formulation used herein is also rather simplistic, counting the cost of a search merely in the number of views processed. The formulation also presupposes a “one-shot” visual system, as opposed to a continual one. The visual search algorithm would necessarily change under a different problem formulation; however, the way spatial relations are included in the solution need not be much changed, we believe.

Acknowledgements

The authors are with the Centre for Autonomous Systems at the Royal Institute of Technology (KTH), Stockholm, Sweden. This work was supported by the SSF through its Centre for Autonomous Systems (CAS), and by the EU FP7 project CogX. K. Sjöö was additionally funded by the Swedish Research Council, contract 621-2006-4520

References

[1] S. Ekvall, D. Kragic, and P. Jensfelt. Object detection and mapping for service robot tasks. *Robotica: International Journal of Information, Education and Research in Robotics and Artificial Intelligence*, 2007.

[2] J. Folkesson, P. Jensfelt, and H. Christensen. The m-space feature representation for slam. *IEEE Transactions on Robotics*, 23(5):1024–1035, Oct. 2007.

[3] T. D. Garvey. *Perceptual strategies for purposive vision.* PhD thesis, Stanford, CA, USA, 1976.

[4] A. Herskovits. *Language and Spatial Cognition*. Cambridge University Press, 1986.

[5] K. Sjö, D. Gálvez López, C. Paul, P. Jensfelt, and D. Kragic. Object search and localization for an indoor mobile robot. *Journal of Computing and Information Technology*, 17(1):67–80, March 2009.

[6] L. Talmy. Force dynamics in language and cognition. *Cognitive Science*, 1988.

[7] A. Torralba, M. S. Castelhano, A. Oliva, and J. M. Henderson. Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search. *Psychological Review*, 113:2006, 2006.

[8] J. K. Tsotsos and K. Shubina. Attention and visual search : Active robotic vision systems that search. In *International Conference on Computer Vision Systems ICVS'07*, Washington, DC, USA, 2007.

[9] L. E. Wixson and D. H. Ballard. Using intermediate objects to improve the efficiency of visual search. *Int. J. Comput. Vision*, 12(2-3):209–230, 1994.

[10] Y. Ye. *Sensor planning for object search*. PhD thesis, 1998.