Semantic Object Search Using Semantic Categories and Spatial Relations between Objects

Patricio Loncomilla, Marcelo Saavedra, and Javier Ruiz-del-Solar

Advanced Mining Technology Center & Dept. of Elect. Eng., Universidad de Chile
{ploncomi,jruizd}@ing.uchile.cl

Abstract. In this work, a novel methodology for robots executing informed object search is proposed. It uses basic spatial relations, which are represented by simple-shaped probability distributions describing the spatial relations between objects in space. Complex spatial relations can be defined as weighted sums of basic spatial relations using co-occurrence matrices as weights. Spatial relation masks are an alternative representation defined by sampling spatial relation distributions over a grid. A Bayesian framework for informed object search using convolutions between observation likelihoods and spatial relation masks is also provided. A set of spatial relation masks for the objects "monitor", "keyboard", "system unit" and "router" were estimated by using images from Label-Me and Flickr. A total of 4,320 experiments comparing six object search algorithms were realized by using the simulator *Player/Stage*. Results show that the use of the proposed methodology has a detection rate of 73.9% that is more than the double of the detection rate of previous informed object search methods.

Keywords: Semantic search, Informed search, Co-occurrence matrix.

1 Introduction

Object search is an important ability for a mobile robot. Some previous work on object search are based on the use of spatial object-place relations, assuming that in a scene the searched object will be readily available to the robot's field of vision. However, there are sets of objects inside a setting that tend to appear near of each other as they have a particular spatial relation, making it possible to infer the existence of an object A, given an object B.

For example, when looking inside offices, it can be noted that the object "keyboard" is often very near the object "monitor". For a human, the ability to deduce that object A tends to be "near", "very near" or "far" from object B is a trivial task, since human beings can learn spatial relations between objects by observing a large number of similar settings. In this paper, we focus on giving robots the ability to find objects using existing semantic relations with other objects such as "near" or "far," in a given environment.

We conducted an exhaustive analysis of spatial relations between real-world objects that share a common use. For this objective, we created a database of spatial co-occurrences where distance values between objects are represented by the linguistic

S. Behnke et al. (Eds.): RoboCup 2013, LNAI 8371, pp. 516–527, 2014.

variables "very near," "near," "far," and "very far". We say that two objects "co-occur" when they appear together in several images showing a particular spatial relation.

The main contribution of this work is the use of convolutions for computing the probability of the presence of an object from positive and negative detections of all the objects on the map in a unified way. There are two secondary contributions. The first one is the representation of spatial relations as spatial relation masks, and the use of basic semantic categories such as "very near," "near," "far," and "very far" for generating basic spatial relation masks that can be combined to generate complex spatial relation masks by using a set of weights named co-occurrence values. The second contribution is the creation of a methodology for computing a co-occurrence matrix associated with a set of basic relation masks from a database of labeled images containing real world objects.

This paper is organized as follows. In Section 2 some related work is presented. In Section 3 the proposed methodology for object search is described. In Section 4 an experimental validation of the methodology is presented. Finally, in Section 5 some conclusions of this work are given.

2 Related Work

The search for objects in a real environment is a very complex task for robots. Garvey in [1] recognized this problem and proposed the idea of indirect search, i.e. searching for another intermediate object that maintains a particular spatial relation with the object being searched for. Wixson et al. materialize the idea of indirect search, demonstrating greater efficiency both theoretically and empirically [2]. But the problem with indirect search is that the spatial relationship between the object sought and the intermediate object does not always exist. Furthermore, the detection of the intermediate object may be more difficult than the detection of the desired/primary object itself. In fact, Ye and Tsotsos demonstrated that the search for an arbitrary object in 3D space is NP-complete [3]. Shubina and Tsotsos propose an algorithm that considers the cost and effect of different actions with different types of prior knowledge and different spatial relations between objects [4]. Kollar et al. perform the search for an object in a known map of the environment, using object-object and object-scene context [5]. They obtain the co-occurrence of the existence of objects in two-dimensional images for learning correlations between object categories, and between objects and place labels (semantic labels such as “kitchen”). These images were taken from the Flickr website, and they do not take the distances between objects into consideration. Viswanathan et al. propose an approach using existing resources: common-sense knowledge of machine learning of object relations [6]. They use marked images from the LabelMe database, designed by Russell et al. [7]. They train an automatic classifier of places based on the presence of the detected objects to infer the probability that the other objects exist, and the kind of place (e.g., kitchen or office) is seen in the setting. Kasper et al. perform a study on spatial relations in three-dimensional images using a Kinect sensor [8]. They created a database using nine different office space settings with a total of 168 objects in 35 object classes. Then, they found the distances between different objects. They also made predictions about the location of unfound objects by detecting their surrounding objects.

Galindo et al. performed a study based on 2D data, combining metric, topological, and semantic aspects on a map [9]. In addition, they proposed a method for learning these semantic representations from sensory data. Vasudevan et al. attempted to create a spatial representation in terms of objects, by encoding typical household objects and doors within a hierarchical probabilistic framework [10]. They used a SIFT [11] based object recognition system and a door detection system based on lines extracted from range scans. They also proposed a conceptualization of different places, based on the objects that were observed inside them.

Aydemir et al. developed a method for object search using explicit spatial relationships between objects in order to perform an efficient visual search [12]. They presented a computational model using several random views to guide the robot's camera to the points where the objects have a high probability of being found by using the spatial-relation term "on" between objects, in an indoor environment since the objects are mostly on horizontal surfaces. The work presented in this paper is an extension and improvement of the models proposed in [8] and [12].

3 Methodology

3.1 Map Update Using Spatial Relation Masks

The proposed methodology is designed for finding an object using informed search, i.e., by using information about other objects, which have a spatial relation with the object to be found. In [12] spatial relations are defined as functions from a space of a pair of poses π_A, π_B of two objects to the interval [0,1], where 1 indicates that the relation is completely fulfilled by these pose combinations, and 0 that the relation does not apply at all:

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

In this work spatial relations will be defined in a probabilistic sense. A spatial relation between two objects is defined as the probability distribution of the pose of the first object given the known pose of the second object:

$$Rel_{A,B}(\pi_A, \pi_B) = p(\pi_A \mid \pi_B) \tag{2}$$

As the relation is treated as a probability distribution, the sum over all possible poses of A for a fixed pose of B is equal to one:

$$\int_{\pi_A} p(\pi_A \mid \pi_B) = 1 \tag{3}$$

If the spatial relation is invariant to translations and rotations, i.e. it only depends on the relative pose $\pi_{A/B}$ of object A with respect to object B, then the expression for the probability can be rewritten as:

$$Rel_{A/B}(\pi_{A/B}) = p(\pi_{A/B}) \tag{4}$$

In our case, the robot is in a two-dimensional space parameterized by using coordinates (x,y). The space is quantized into squared cells with size k, and parameterized by using indexes (i,j). By using the index notation, a spatial relation can be written as:

$$R_{A/B}(i,j) = K_{norm} * Rel_{A/B}(ki,kj) \tag{5}$$

$$\sum_i \sum_j R_{A/B}(i,j) = 1 \tag{6}$$

where K_{norm} is a normalizing constant.

The term $p(a_{i,j})$ represents the probability that the center of the main object A is in the cell (i,j). Both positive and negative observations z_A provide valuable information for the object search process, and can be used to compute an updated probability $p(a_{i,j}|z_A)$. The probability $p(a_0)$ is treated as a special case, and it represents the probability of the object being outside the search region.

Positive detections z_A=*true* provide information about the places where the object has high probability of being, by means of a likelihood $p(z_A$=*true*$|a_{i,j})$, which is defined over the cell (i,j). The likelihood has a high value over the cell where the object was detected, and a low value in the other cells. Negative detections z_A=*false* provide information $p(z_A$=*false*$|a_{i,j})$ about the cells where the objects have low probability of being, which are those cells visible from the current viewpoint that have a low probability of containing the object.

The problem addressed in this work is to find a main object, A, by moving the robot appropriately. The robot search process is applied until the main object, A, is found. In consequence, the search process includes only negative detections of the main object, A, before the object is found. Two cases are considered:

$$p(a_{i,j} \mid z_A = false) = \frac{p(z_A = false \mid a_{i,j}) p(a_{i,j})}{p(a_o) + \sum_{i,j} p(z_A = false \mid a_{i,j}) p(a_{i,j})} \tag{7}$$

$$p(a_0 \mid z_A = false) = \frac{p(a_0)}{p(a_o) + \sum_{i,j} p(z_A = false \mid a_{i,j}) p(a_{i,j})} \tag{8}$$

The secondary object, B, can produce positive and negative detections z_B, which can be used to compute an updated probability:

$$p(a_{i,j} \mid z_B) = \frac{p(z_B \mid a_{i,j}) p(a_{i,j})}{p(z_B \mid a_o) p(a_0) + \sum_{i,j} p(z_B \mid a_{i,j}) P(a_{i,j})} \tag{9}$$

$$p(a_0 \mid z_B) = \frac{p(z_B \mid a_o) p(a_0)}{p(z_B \mid a_o) p(a_0) + \sum_{i,j} p(z_B \mid a_{i,j}) P(a_{i,j})} \tag{10}$$

The terms $p(z_B|a_{i,j})$ and $p(z_B|a_0)$ will be called cross-likelihoods, as they relate the detection of a secondary object, B, with the presence of the main object, A, on the map.

These probabilities can be derived by considering probabilities $p(b_{u,v})$ for the presence of a secondary object, B, at locations (u, v) in the grid:

$$p(z_B \mid a_{i,j}) = \sum_u \sum_v p(z_B \mid b_{u,v}) p(b_{u,v} \mid a_{i,j}) \tag{11}$$

$$p(z_B \mid a_0) = \sum_u \sum_v p(z_B \mid b_{u,v}) p(b_{u,v} \mid a_0) \tag{12}$$

The term $p(b_{uv}|a_0)$ is considered a constant over (u,v) whose sum has a value of 1 because B is supposed to be on the map. The term $p(b_{u,v}|a_{i,j})$ corresponds to the spatial relation between the main object, A, at location (i, j) and a secondary object, B, at location (u, v). By replacing this term with the spatial relation $R_{B/A}$, there is no need for storing a map for the secondary object; only the map for the main object and the likelihoods of the detections of the secondary object are needed:

$$p(z_B \mid a_{i,j}) = \sum_u \sum_v p(z_B \mid b_{u,v}) R_{B/A}(u-i, j-v) \tag{13}$$

$$p(z_B \mid a_0) = \frac{1}{n_U n_V} \sum_u \sum_v p(z_B \mid b_{u,v}) \tag{14}$$

where $n_U n_V$ is the size of the map.

Equation (13) can be implemented as a convolution in the (i,j) space between a likelihood image and a mask $R_{B/A}(i,j)$ describing the spatial relation between the main and secondary objects, which will be named a *spatial relation mask*:

$$p(z_B \mid a_{i,j}) = p(z_B \mid b_{i,j}) * R_{B/A}(i, j) \tag{15}$$

The proposed system is highly versatile because any spatial relation can be represented by an appropriate mask. It must be noted that extra secondary objects can be added to the system by creating additional spatial relation masks. In case these relations are chained, as an example object A is near B, and object B is near C, then the mask of the chained relation can be obtained by convolution of the original masks:

$$P(z_C \mid a_{i,j}) = P(z_C \mid b_{i,j}) * R_{B/A}(i, j) \tag{16}$$

$$P(z_C \mid a_{i,j}) = P(z_C \mid c_{i,j}) * R_{C/B}(i, j) * R_{B/A}(i, j) \tag{17}$$

$$\Rightarrow R_{C/A} = R_{C/B} * R_{B/A} \tag{18}$$

A path for searching for the object can be created by generating optimal viewpoints at each iteration. The optimal viewpoint is generated from a set of k random poses reachable in a fixed time, and selecting the one that maximizes the probability of finding the object in the visible area, as shown in [8]:

$$\arg\max_{k=1..N} \sum_{i=1}^{n} \sum_{j=1}^{n} p(a_{i,j}) V(a_{i,j}, k) \tag{19}$$

where N is the number of candidate poses, and $V(a_{i,j},k)$ is defined as:

$$V(a_{i,j},k)=\begin{cases}1, & \text{if } a_{i,j} \text{ is inside the k}^{th} \text{ view cone}\\ 0 & \text{otherwise}\end{cases} \tag{20}$$

The navigation algorithm has an impact on the reliability of the object search algorithm. For generating random poses, random sets of parameters for the navigation algorithm must be selected. Simple navigation strategies like going forward and rotating have an easily computable navigation time for a given final pose, and more complex navigation algorithms are not easily parameterized, then there is a trade-off between simplicity of the navigation algorithm (which allows simple and accurate generation of trajectories), and complex navigation approaches (which enable the robot to reach better poses in a given time). In the experiments reported in Section 4 we choose to use a simple navigation strategy, which every time selects the execution of a composed sequence of movements consisting on one initial rotation, followed by a translation and a final rotation. The parameter space associated to this sequence of movements is explored randomly, and the best sequence is chosen as the one that maximize the probability of finding the searched object.

For computing the convolutions, a way of generating appropriate masks from examples of real-world images is needed. This will be explained in the following section.

3.2 Creating Spatial Relation Masks from Co-occurrences

In this section, a procedure for approximating a complex spatial relation as a weighted sum of basic spatial relations is presented. Each basic spatial relation corresponds to a semantic category meaningful to humans. In this work, we focus on four simple spatial relations: "very near" (VN), "near" (N), "far" (F), and "very far" (VF). The use of these spatial relations is useful as it enables the system to estimate a set of basic probability distributions from samples of relative positions of the objects in the real world. The masks for each of the spatial relations are defined in two versions, *hard masks* and *soft masks*. Hard masks are defined by two thresholds and have a rectangular profile, while soft masks are defined by four numbers and have a trapezoid-shaped profile. Each basic mask is normalized to sum one over all of the cells on the map, thus a normalization constant is added to the formulas

Equations for hard masks are defined in equations (21) to (25).

$$R_{B/A\,hard}(i,j;a_1,a_2)=K*\begin{cases}1 & a_1\le\sqrt{(ki)^2+(kj)^2}<a_2\\ 0 & other\end{cases} \tag{21}$$

$$R^{VN}_{B/A\,hard}(i,j)=R_{B/A\,hard}(i,j;0,u_1) \tag{22}$$

$$R^{N}_{B/A\,hard}(i,j)=R_{B/A\,hard}(i,j;u_1,u_2) \tag{23}$$

$$R^{F}_{B/A\,hard}(i,j)=R_{B/A\,hard}(i,j;u_2,u_3) \tag{24}$$

$$R^{VF}_{B/A\,hard}(i,j)=R_{B/A\,hard}(i,j;u_3,\infty) \tag{25}$$

Equations for soft masks are similar, but have a smooth transition between values 0 and 1, that is regulated by a gap parameter δ. The equations for soft masks are shown in equations (26) to (30).

$$R_{B/Asoft}(i,j;a_1,a_2,a_3,a_4) = K * \begin{cases} \dfrac{\sqrt{(ki)^2+(kj)^2}-a_1}{a_2-a_1} & a_1 \le \sqrt{(ki)^2+(kj)^2} < a_2 \\ 1 & a_2 \le \sqrt{(ki)^2+(kj)^2} < a_3 \\ \dfrac{a_4-\sqrt{(ki)^2+(kj)^2}}{a_4-a_3} & a_3 \le \sqrt{(ki)^2+(kj)^2} < a_4 \\ 0 & other \end{cases} \tag{26}$$

$$R^{VN}_{B/A\,soft}(i,j) = R_{B/A\,soft}(i,j;0,0,u_1-\delta,u_1+\delta) \tag{27}$$

$$R^{N}_{B/A\,soft}(i,j) = R_{B/A\,soft}(i,j;u_1-\delta,u_1+\delta,u_2-\delta,u_2+\delta) \tag{28}$$

$$R^{F}_{B/A\,soft}(i,j) = R_{B/A\,soft}(i,j;u_2-\delta,u_2+\delta,u_3-\delta,u_3+\delta) \tag{29}$$

$$R^{VF}_{B/A\,hard}(i,j) = R_{B/A\,hard}(i,j;u_3-\delta,u_3+\delta,\infty,\infty) \tag{30}$$

In this work, basic masks defined by a circle of radius u_1 in the case of "very near", a circular ring of radii u_1 and u_2 in the case of "near," a circular ring of radii u_2 and u_3 in the case of "far," and a circular ring of internal radius u_3 and an external radius that cover the whole map in the case of "very far". The radius values are selected by considering statistics of the distances between objects A and B, and by modeling their selection process as a classification problem. Thus, the optimal radius value between two categories, e.g., "near" and "far", is the one that generates the same mean classification error in both classes.

A complex mask can be created as a weighted sum of basic hard or soft masks:

$$\begin{aligned} R_{B/A}(x,y) = {} & C^{VN}_{B/A} R^{VN}_{B/A}(x,y) + C^{N}_{B/A} R^{N}_{B/A}(x,y) + \\ & + C^{F}_{B/A} R^{F}_{B/A}(x,y) + C^{VF}_{B/A} R^{VF}_{B/A}(x,y) \end{aligned} \tag{31}$$

An example of a complex mask sampled over a grid with pixel size 0.1[m] is shown in Figure 1.

The four coefficients $C^{VN}_{B/A}$, $C^{N}_{B/A}$, $C^{F}_{B/A}$ and $C^{VF}_{B/A}$ are called co-occurrences because they indicate the relative frequency of occurrence of a pair of objects for each spatial relation. They can be constructed from samples of positions of both objects by computing the number of occurrences of each basic spatial relation. If a set of samples is divided into basic semantic categories and the count is n_{VN} for "very near," n_N for "near," n_F for "far," and n_{VF} for "very far," the co-occurrences can be computed by using equations (32) to (35):

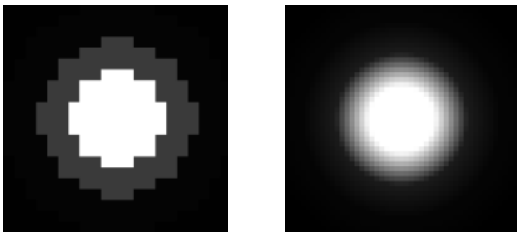
$$C^{VN}_{B/A} = \frac{n_{VN}}{n_{VN}+n_N+n_F+n_{VF}} \tag{32}$$

$$C^{N}_{B/A} = \frac{n_N}{n_{VN}+n_N+n_F+n_{VF}} \tag{33}$$

$$C_{B/A}^{F} = \frac{n_F}{n_{VN} + n_N + n_F + n_{VF}} \tag{34}$$

$$C_{B/A}^{VF} = \frac{n_{VF}}{n_{VN} + n_N + n_F + n_{VF}} \tag{35}$$

A co-occurrence matrix is the set of co-occurrence values of two or more objects with a particular spatial relation.



(a) (b)

Fig. 1. (a) Hard complex mask composed by summing hard masks "very near," "near," and "far," sampled over a grid. (b) The equivalent soft complex mask (see main text for details).

4 Results

4.1 Experimental Setup

In order to characterize the proposed object search methodology and to compare it with existing methodologies, a simulation environment that uses data of real-world objects was developed. We believe that a simulation environment is an appropriate tool to evaluate object search algorithms, because it allows obtaining repeatable experiments and to better quantify the performance of the different methodologies.

Experiments were performed on maps containing four objects, by using the *Player/Stage*, a robot simulation tool [13]. Each map contains a main object, named A (monitor), to be searched for, and three secondary objects named B (keyboard), C (system unit), and D (router). The size of each map is 6[mt] x 6[mt]. A total of 20 maps were created by picking a random position for the main object, A, and then picking a random position for the objects B, C, and D following a distribution that represents the co-occurrences shown in Table 1. The 20 maps are used to perform the experiments, each map being used the same number of times as the others. A laser sensor from *Player/Stage* is used for avoiding collisions. Observations of the objects are obtained by using a fiducial sensor included in Player/Stage that is able to measure the position of a detected object inside a view cone whose size depends on the object: the main object, A, can be detected up to 1[mt], and the secondary objects, B, C, and D can be detected up to 2[mt].

In each experiment the goal is to find the main object A before 1,500 views have been processed. In the search process, only negative detections of the main object need to be processed because a positive detection causes the search process to finish.

Six algorithms of object search are compared on the same set of maps. The first 4 correspond to different variants of the proposed methodology, the fifth algorithm is the one proposed by Aydemir in [12], and the sixth one corresponds to the baseline algorithm where no information from secondary objects is used. The algorithms are:

1. *Informed search with convolutions using positive and negative information with hard masks*: A probability map $P(a_{i,j})$ for the main object A is estimated by using positive and negative detections of objects B, C, and D, negative detections of object A, and spatial relation masks. Then object A is searched by finding viewpoints that maximize the probability of containing it. The relation masks $R_{A/B}$, $R_{A/C}$ and $R_{A/D}$ needed for updating the probability map $P(a_{i,j})$ from detections of secondary objects are constructed by using co-occurrence matrices.
2. *Informed search with convolutions using only positive information with hard masks*: Similar to algorithm 1, but in this case only positive detections of objects B, C, and D are used.
3. *Informed search with convolutions using positive and negative information with soft masks*: Similar to algorithm 1, but in this case soft spatial relation masks are used.
4. *Informed search with convolutions using only positive information with soft masks*: Similar to algorithm 2, but in this case soft spatial relation masks are used.
5. *Informed search using particles*: The algorithm of Aydemir [12] is used for constructing a probability map $P(a_{i,j})$ for object A by using negative detections of that object. Then object A is searched for by finding viewpoints that maximize the probability of containing A. When a secondary object is detected, a set of particles is generated around the detection inside the current view cone, and the ones which fulfill the spatial relation computed from Table 1 are used to select the next optimal viewpoint. A spatial relation is considered fulfilled when its current value is equal or greater than half of its maximum possible value.
6. *Uninformed search*: A probability map $P(a_{i,j})$ for object A is estimated by using negative detections of that object, then object A is searched for by finding viewpoints that maximize the probability of containing A. This is the baseline algorithm used in [12], and no information from secondary objects is used.

4.2 Creation of Co-occurrence Matrices

A set of 243 images from LabelMe [7] and captions of photos on the Flickr website were used for generating co-occurrence matrices. In each image, instances of the objects "monitor," "system unit," "keyboard," and "router" were labeled. As the sizes of the objects and the parameters of the camera are known, it is possible to compute the pose of each object in space. Several instances of the objects on the set of images and their poses were used to construct co-occurrence matrices for the categories "very near," "near," "far," and "very far" for each of the objects with respect to the others.

Given the poses of a pair of objects, a distance was computed and used for selecting whether the sample belongs to the categories "very near," "near," or "far". If an object is detected alone in an image, the sample belongs to the category "very far". Only the depth and horizontal axis were used to compute the distances, as differences in the vertical direction do not affect the position of the object when it is transformed onto the 2D grid. The statistics of the distances between the object "monitor" and the objects "keyboard," "system unit," and "router," as well as the delimitation between the basic spatial relations are shown in Figure 2. The final co-occurrences for the object, *Monitor,* as the main object are shown in Table 1.

The basic hard and soft masks are defined by equations (21)-(25) and (26)-(30), respectively. The thresholds that separate the categories very near, near, far and very far are u_1=60[cm], u_2=100[cm] and u_3=150[cm]. The gap parameter for the soft masks is d=20[cm].

Table 1. Final co-occurrences of objects around the object "monitor"

Main object Monitor	Secondary objects		
Semantic categories	*keyboard*	*system unit*	*router*
Very Near	0.773	0.178	0.061
Near	0.143	0.491	0.151
Far	0.046	0.258	0.485
Very Far	0.038	0.074	0.303

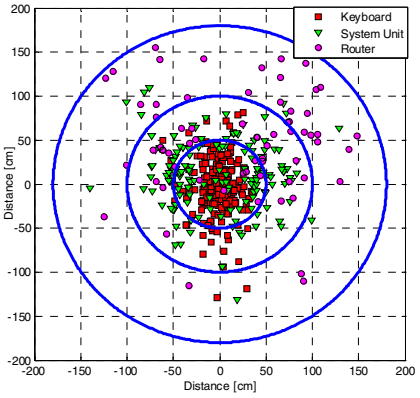


Fig. 2. Relative poses between object "monitor" and objects "keyboard," "system unit," and "router". Thresholds for basic spatial relations are shown as circles.

4.3 Experiments

A total of 4,320 experiment trials were executed for comparing the six search algorithms explained in Section 4.1. For each algorithm, a total of 720 experiments considering different maps and different initial robot poses were executed. Each experiment is considered successful if the object is found before 1,500 sensor frames. This considers that all algorithms are able to run at 15 fps, which is the frame rate used in the simulations. The results from the experiments are shown in Table 2.

From the results of the experiments, it is evident that the algorithms that use convolutions for integrating information about secondary objects into the probability distribution of the main object have the best performance. This happens because the integrated information can be used several frames after the moment when the object was seen. The integration of positive detections of secondary objects using hard masks improves the detection rate from 28.9% up to 49.6%. The use of soft masks is also a factor that improves the detection rate up to a 53.2%. Finally, the use of

positive and negative information, in addition to soft masks, generates an important improvement in the detection rate that rises up to a 73.9%. The integration of information using masks, the comparison between different kind of masks and the ability of using negative information about secondary objects are all contributions of this work. Aydemir's particle based informed search algorithm [12] performs better than the baseline; however, the best algorithm described in this paper has a detection rate that is more than the double of the detection rate of Aydemir's algorithm.

The methods can be optimized by observing that the cross-likelihood images are constant except on the detection area in the case of positive detections, and on the view cone in the case of negative detections. Then, focalized convolutions can be applied on these areas. Informed search using positive and negative information scales linearly with the number of total secondary objects. Informed search using only positive information scales linearly with the number of observed secondary objects. As the mean amount of detected objects is low, the methods that use only positive information run as fast as uninformed search the most of the time. Both kind of methods scale quadratically with the size of the mask when using focalized convolutions, and they scale linearly with the size of the mask when using convolutions with the full map. Time of focalized convolutions do not depend on the size of the map. In the experiments, a small map (100x100) and mask (32x32) were used for representing a 6m x 6m environment and the processing time of informed search is lower than the processing time of object detection algorithm.

Table 2. Results from the experiments comparing six variants of object search algorithms. In each variant, a total of 720 experiments were performed. Each experiment is successful if the main object is found before 1,500 sensor frames. DR: Detection Rate.

Algorithm	Number of searches	Successful searches	DR
Informed search using positive and negative information and hard masks	720	496	68.9%
Informed search using only positive information and hard masks	720	357	49.6%
Informed search using positive and negative information and soft masks	720	532	73.9%
Informed search using only positive information and soft masks	720	383	53.2%
Informed search using particles(Aydemir et al. [12])	720	248	34.4%
Uninformed search	720	208	28.9%

5 Conclusions

In this work, a novel methodology for performing informed search of objects was proposed and tested. The methodology is based on integrating information provided by secondary objects into the probability distribution of the main object to be found. Spatial relations between objects are estimated by using a set of basic spatial relations which are mixed by using co-occurrence values as weights. Six algorithms of object search were compared by using spatial relations estimated from real-world data by

performing a total of 4,320 simulations in *player/stage*. The results show that the detection rate of the search process increases from 28.9% to 73.9% when integrating positive and negative detections from the secondary objects into the probability distribution of the main object using soft masks. The integration of positive and negative detections of secondary objects, as well as the use of soft masks increases the detection rate. The obtained detection rate is more than the double of the one obtained by previous informed search algorithms.

Future work includes the creation of a full 3D model of the informed search system, the management of false detections, the comparison with extra object search methods, and the realization of experiments with a real robot for validating the results in the real world.

Acknowledgments. This work was partially funded by FONDECYT under Project Number 1130153.

References

[1] Garvey, T.D.: Perceptual strategies for purposive vision, Technical report, SRI International, vol. 117 (1976)
[2] Wixson, L., Ballard, D.: Using intermediate object to improve efficiency of visual search. Int. J. Comput. Vis. 18(3), 209–230 (1994)
[3] Ye, Y., Tsotsos, J.K.: Sensor Planning for 3D Object Search. Computer Vision and Image Understanding 73-2, 145–168 (1999)
[4] Shubina, K., Tsotsos, J.: Visual search for an object in a 3d environment using a mobile robot. Computer Vision and Image Understanding 114(5), 535–547 (2010)
[5] Kollar, T., Roy, N.: Utilizing object-object and object-scene context when planning to find things. In: Proc. of the 2009 IEEE Int. Conf. on Robotics and Automation, ICRA 2009 (2009)
[6] Viswanathan, P., Meger, D., Southey, T., Little, J.J., Mackworth, A.: Automated Spatial-Semantic Modeling with Applications to Place Labeling and Informed Search. In: Proc. Canadian Conf. on Computer and Robot Vision (2009)
[7] Russell, B., Torralba, A., Murphy, K., Freeman, W.: Labelme: A database and web-based tool for image annotation. International Journal of Computer Vision 77, 157–173 (2008)
[8] Kasper, A., Jäkel, R., Dillmann, R.: Using spatial relations of objects in real world scenes for scene structuring and scene understanding. In: Proc.15th Int. Conf. on Advanced Robotics, ICAR 2011, Tallinn, Estonia (2011)
[9] Galindo, C., Saffiotti, A., Coradeschi, S., Buschka, P., Fernandez-Madrigal, J.A., Gonzalez, J.: Multi-hierarchical semantic maps for mobile robotics. In: Proc. 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS 2005, pp. 2278–2283 (2005)
[10] Vasudevan, S., Gachter, S., Nguyen, V., Siegwart, R.: Cognitive maps for mobile robots-an object based approach. Robot. Auton. Syst. 55, 359–371 (2007)
[11] Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. Int. Journal of Computer Vision 60, 91–110 (2004)
[12] Aydemir, A., Sjöö, K., Jensfelt, P.: Object search on a mobile robot using relational spatial information. In: Proc. 11th Int Conf. on Intelligent Autonomous Systems, IAS 2011 (2010)
[13] The Player Project, `http://playerstage.sourceforge.net/`