Active Visual Search by a Humanoid Robot

Francois Saidi, Olivier Stasse and Kazuhito Yokoi
ISRI/AIST-STIC/CNRS Joint Japanese-French Robotics Laboratory (JRL)
Central 2, 1-1-1 Umezono, Tsukuba, 305-8568 JAPAN
Email: francois.saidi,olivier.stasse,kazuhito.yokoi@aist.go.jp

***Abstract*— This paper presents a framework for a visual search behavior of a 3D object in a 3D environment performed by a HRP-2 humanoid robot. The object search falls in the field of sensor planning and is formulated as an optimization problem. The goal is to maximize the target detection probability while minimizing the energy/distance and time to achieve the task. This paper propose some natural constraints based on specificities of the humanoid robot and on the characteristics of the recognition system to reduce the dimension of the problem. The paper presents simulation results of an object search behavior using the HRP-2 robot.**

I. INTRODUCTION

A. The visual search behavior

Object search is a very common task we perform each time we need an object. Humanoid robots are multipurpose platforms and will need to use generic tools to extend their capacities. It must thus be able to look for objects, to localize and use them. A search behavior would be a great improvement in humanoid autonomy and a step forward toward their rise outside laboratories.

Before starting a search behavior, the robot needs a model of the desired object. This model could be provided by an external mechanism, but a humanoid has all the required abilities to build that model by its own. An undergoing project in our laboratory, called the "Treasure hunting" aim at integrating in a unique cycle, the model building of an unknown object, and the search for that object in an unknown environment. With such a combined skill, the robot may incrementally build a knowledge of its surrounding environment and the object it has to manipulate without any a-priori models. Latter the robot would be able to find and recognize that object. The time constraint is crucial, as a reasonable limit has to be set on the time an end user can wait the robot to achieve its mission. This paper will focus on the search behavior and we assume that the object model is already created.

B. Problem statement and contributions

Object search is a sensor planning problem which is proven to be NP-complete [1] thus a heuristic strategy is needed to overcome that task. Because of the limited field of view, the limited depth, the lighting condition, the recognition algorithm limitation, and possible occlusion, many images from different point of view are necessary to detect and locate a given object. The knowledge of the target position is represented by a discrete presence probability [2]. A rating function to evaluate the interest of a potential next view must be created and optimized at each sensing step. The rating function will analyze the theoretical field of view for a given configuration according to various criteria defined further in this paper. Such a function is costly and thus must be used as less as possible to evaluate a configuration interest.

In [3], we introduce the concept of *Visibility Map* a statistical accumulator in the sensor configuration space which takes into account the characteristics of the recognition system to constrain the sensor configuration space and avoid unnecessary call to the rating function. The present paper proposes an extention of the visibility map and exposes a process to retrieve interesting configuration out of that map (section II-D).

C. Related works

Few works on active 3D object search are available, fortunately the sensor planning research field provides us with some hints.

Wixon [4] uses the idea of indirect search (in which one first finds an object that commonly has a spatial relationship with the target, and then restrict the search in the spatial area defined by that relationship) he proposes a mathematical model of search efficiency, which shows that indirect search can improve the search.

Works done by Ye and Tsotsos [2] tackle the field of sensor planning for 3D object search. The search agent's knowledge of object location is encoded as a discrete probability density which is updated after each sensing action performed by the detection function. The detection function uses a simple recognition algorithm, and all factors which influence the detection ability such as imaging parameters, lighting condition, complexity of the background, occlusions etc. are included in the detection function value by averaging experimental results done under various conditions. The vision system uses one pan tilt zoom camera and a laser range finder to build a model of the environment. The search is not really 3D as, the object is recognized using a 2D technique, and the height of the camera is fixed.

Works by Sujan [5] are not focused on object search but on accurate mapping of unknown environment by the mean of sensor planning. The author propose a model based on iterative planning, driven by an evaluation function based on Shannon's information theory. The camera parameter space is explored and each configuration is evaluated according to the evaluation function. No computational timing tests are

provided, but the algorithm seems to focus on configurations which are close to obstacles or to unknown areas to improve the algorithm efficiency, this latter constraint will be formalized with the notion of visibility map introduced in II-C.

The operational research community [6] has extensively studied the problem of optimal search, they came up with interesting theoretical results on search effort allocation which served as a basis for Tsotsos's work.

The Next Best View (NBV) research field [7] studied the sensor planning problem mainly for C.A.D. model building. These works, although sharing some commom aspects with the present topic, rely on the fundamental assumption that the object is always in the sensor field.

II. Constraint on the camera parameters space

A. Specificities of humanoid approach

Specificities of the HRP-2 humanoid robot must be taken early into account in the search behavior analysis.

The walking pattern generator provided by [8] constrain the waist motion on a plane, as a consequence the head is also restricted on a plane called the visual plane. During the walk, the robot point of view oscillate around that plane with an amplitude of 2cm which falls inside the resolution used by environment model. This constrain will be removed in a future work as a new pattern generator is available [9] which accepts large perturbations on the waist height.

Unlike [5], the visual sensor, which is located in the head of the robot, is subjected to stability constraint. In this work we don't consider robot postures in which the head of the robot goes over obstacles, thus the sensors configuration space is restricted by the 2D projection of obstacles on the visual plane. Moreover, we introduce a safety margin around obstacles in which sensor placement will not be evaluated.

These remarks on humanoids specificities provide natural constraints on the sensor configuration space. Other constraints due to the stereoscopic sensor and the recognition algorithm will be discussed in II-C.

B. Model of the recognition system

All recognition algorithms have some restrictions regarding the imaging condition (lighting, occlusion, scale...). One of the main assumption that can be easily controlled by active vision is the scale limitation: the smallest scale at which the object can still be recognized constitute a maximum distance limit for the detection algorithm (R_{max}). It is also suitable to have a sensor configuration in which the whole object is projected inside the image in order to maximize the number of imaged features, this imposes a lower limit for the sensor distance to the object (R_{min}). Without any loss of generality regarding the recognition algorithm, we can assume that these bounding values (R_{min} and R_{max}) are determined theoritically or experimentally during the model building and are stored with the object model. These limit values will be used to further constrain the sensor parameters to improve optimization time.

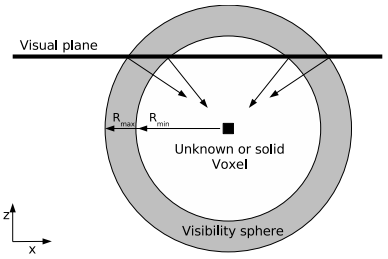

Fig. 1. Visibility sphere for a given 3D point

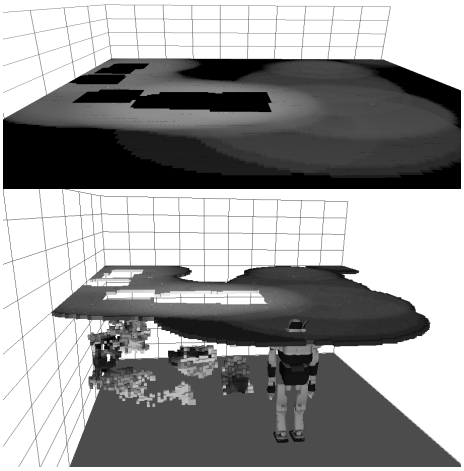

Fig. 2. This visibility map is only computed for reconstructed solid points (gray points under the plane). Each point is creating a visibility sphere around it. Lighter area on the plane represent configurations in which the solid points can be well imaged

C. The visibility map

To take into account the limitation of the recognition algorithm, and to restrict the optimization to area of interest, we use the concept of visibility sphere which represents the configuration set of the stereoscopic head in which a particular 3D point can be well recognized by a given recognition algorithm. This sphere is created using R_{min} and R_{max} defined in II-B. Figure 1 shows a 2D representation of the visibility sphere when a unique solid point is considered.

The configuration space of the stereoscopic head has initially 6 DOF but because the robot motion is constrained on the z axis, and the roll parameter (rotation around the line of sight) has a small influence on the visible area, only 4 DOF are considered. The sensor configuration space parameters are discretisized using the same resolution as the occupancy grid for x and y (5 cm). Whereas for pan and tilt, a resolution of half the stereoscopic field of view value, which is 33 degrees horizontally and vertically, is used.

For each solid or unknown point, the visibility sphere according to R_{min} and R_{max} values is computed and the contributions of all solid and unknown points are summed up in an accumulation map. The visibility map is then constrained on the z axis by computing its intersection with the visual plane. The figure 2 shows a 2D projection of the 4D visibility map.

In a previous work, the visibility map was computed

on a 2.5D projection of the environment, this solution although computationally efficient, did not take into account an important part of the potentially visible points of the environment. Moreover, this technique did introduce a skew in the visibility map creating false interesting configurations. In the current paper, we now compute the visibility map for all boundary points (unknown or solid voxels with an empty neighbor). This new approach increases the computation time of the visibility map but takes into account all the visible 3D surface made of unknown or solid points of the environment. This computational overload can be reduced by some algorithmic improvement discussed in IV-B.

D. Local maxima extraction

The visibility map can be seen as a 4D, gray values map:

- The value of each configuration in the visibility map is called the visibility of the configuration. A candidate is a configuration which has a non zero visibility.
- The set of candidate which have the same x and y parameter is called a cluster (the cluster visibility is the sum of all its candidates visibility). Figure 2 shows in fact the clusters of the visibility map.

In order not to introduce unuseful candidates, the visibility map is only computed in the reachable area (area of the visual plan which is connected to the current sensor position). Nevertheless, a pretreatment of the visibility map is necessary to reduce the number of configurations to send to the rating function.

The basic idea of the treatment is to provide the evaluation function with configurations which respect certain criteria:

- For each configuration, a certain amount of points of interest must be visible
- Points of interest must be seen under imaging condition which allow a reliable recognition
- Configuration must have a low coupling (their view field must weakly intercept)
- The set of all configurations must partition the visible space

The coupling inside the same cluster is low because a change in the pan tilt parameter will bring a lot of new information in the field of view. On the other hand, a change in the x,y parameters will most likely produce a small change in the field of view. A local maxima extraction of the visibility map based on a window with different size for the rotation and translation parameters will output the 'locally best' configurations for which a reasonable amount of point is visible. A small size is used for the pan and tilt parameter, reflecting the fact that configurations with close orientation value are weakly coupled. A larger window size is used on the translation parameters. In this paper we use a window of size 3 for rotation and 9 for translation in the discreet parameter space.

The greedy exploration of sensor's parameter space is constrained to the local maxima of the visibility map. An interesting feature of the visibility map comes from the fact that solid and unknown points are treated the same way, and generate their visibility sphere, thus suitable configurations for exploring unknown areas are also created.

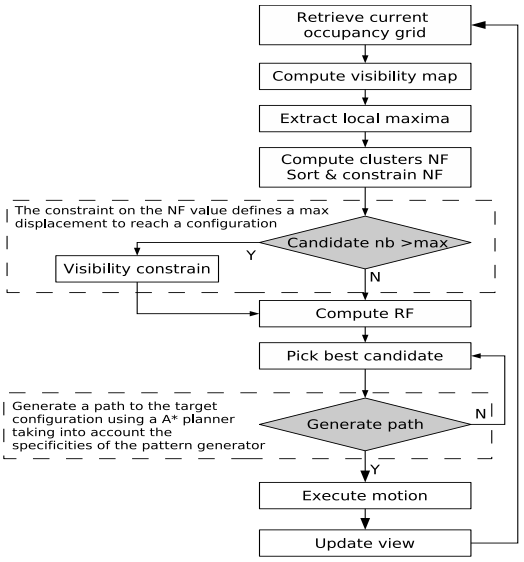

Fig. 3. Flowchart of the next view selection

Next section will present the overall algorithm.

III. Algorithm

A. Overview

The flowchart of the next best view selection process is depicted in figure 3. When a new world model is available, the corresponding visibility map is computed and the local maxima extraction is performed providing a candidate list. The following sections describe the different steps of the next view selection as well as the formulation of the rating function, more details can be found in [3].

B. The probability world map

A discrete occupancy grid is generated by the stereoscopic sensor of the robot (figure 4). Localization is done through a SLAM process [10] which merges odometric information provided by the walking pattern generator and visual information to provide accurate positioning. The target presence is represented by a discrete probability distribution function p. Since this probability will be updated after each recognition action, it is a function of both position and time. $p(v_i, t)$ represents the probability that the voxel v_i is a part of the target. For a given camera configuration c,

$$P(c) = \sum_{\Psi} p(v_i, t), \tag{1}$$

represents the probability that the object is inside the current field of view Ψ. The field of view takes into account occlusions for already mapped obstacles as well as the depth of field.

C. The rating function

The rating function must evaluate the interest of a given configuration according to different criteria:

1) the probability of detecting the object: the detection probability (DP),
2) the new area of the environment that will be seen: the new information (NI),
3) the cost in time/energy to reach that configuration: the motion cost (MC).

The DP, NI and MC are combined in the rating function:

$$RF = \lambda_{DP} \cdot DP + \lambda_{NI} \cdot NI - \lambda_{MC} \cdot MC, \qquad (2)$$

where λ_{DP}, λ_{NI} and λ_{MC} are scaling factor to balance the contribution of each member of the rating function. This function will be optimized to select the next view.

The weights selection depends on the current strategy of the search:

- a high λ_{NI} will support a wide exploration of the environment,
- a high λ_{DP} will support a deep search of each potential target.

The following sections will describe the different part of the rating function.

D. The detection probability

Resolution studies done by [11] provide a characterization of the stereoscopic sensor of the robot. The resolution factor $\rho(v_i)$ which gives the resolution at which each voxel is perceived is used to modulate the recognition likelihood. This function is defined on the field of view Ψ and has 3 parameters (θ, δ, l).

From equation 1 we define the detection probability (DP) for a given camera parameter c as:

$$DP(c) = \sum_{\Psi} p(v_i, t)\rho(v_i). \qquad (3)$$

E. The new information

The new information (NI) concept already introduced by [12] and [5] is also used in the overall configuration rating process but with a different formulation. In these works, the expected information evaluation for a given sensor configuration did not take into consideration the occlusion problem. The only occlusion that was considered is the one created for already known obstacles. In [3] we proposed a novel formulation of the information measurement which integrates an occlusion prediction. With such a formulation we could maximize the expected information while minimizing the likelihood of occlusion.

In order to have a measurement on the possible occlusion in unmapped areas, we evaluate both the minimum and maximum expected information:

- The minimum predicted information (I_{min}), in which all unknown voxels are expected to be solid and thus causes high occlusion which, in return, will decrease the available information.
- The maximum expected information (I_{max}), in which all voxels are expected to be empty and for which all unknown voxels will reveal information.

$$NI = \alpha_{avg} \cdot \frac{I_{max} + I_{min}}{2 \cdot N} + \alpha_{err} \cdot \frac{I_{min}}{I_{max}}, \qquad (4)$$

where N is the total number of voxel in the field of view when there is no occlusion, α_{avg} and α_{err} are the coefficient for the expected average and error ($I_{min} \leq I_{max}$) and $NI = 0$ when $I_{max} = 0$. With this formulation maximizing NI, will on one hand, maximize the average expected information $\frac{I_{max}+I_{min}}{2 \cdot N}$, while on the other hand, minimize the error on the prediction $\frac{I_{min}}{I_{max}}$.

F. The motion cost

In addition to maximizing the NI and DP, it is also interesting to minimize the distance to travel to reach the configuration. An Euclidean metric in the configuration space of the sensor with individual weights on each DOF, is used to define the motion cost (MC). Moreover to take into account obstacles, we integrate a navigation function based on a 2D projection of the occupancy grid to evaluate the motion cost on the x and y parameters of the sensor.

$$MC = \alpha_{NF} \cdot NF(x, y) + \sqrt{\alpha_p (p' - p)^2 + \alpha_t (t' - t)^2}, \qquad (5)$$

In this paper, the pan-tilt (p,t) parameters have a low weight (α_p,α_t) whereas x and y have a higher weight (α_{NF}) reflecting the fact that a change of x and y is achieved by moving the whole robot which takes more time and energy than moving only the head.

Next section presents the optimization of this rating function in order to determine the next sensor configuration.

G. Candidates examination

The local maxima extraction presented in section II-D provides us with a list of candidates. This candidates list could directly be sent to the rating function, but for efficiency reasons the different parts of the rating function are evaluated separately starting with the less computationally expensive part, the motion cost. The navigation function (section III-F) $NF(x, y)$ is computed for all positions. A distance criteria is first applied to constrain the candidates inside a neighborhood around the current robot position (a typical value is 2m, which guaranties that the next view will be within a 2m distance).

If the candidates are still too numerous, a visibility constrain is applied and the best candidates are taken (i.e. candidates wich recived to maximum amount of votes). The number of candidates that can be sent to the rating function depends on the reaction time we want to achieve an on the state of the robot (i.e. when the robot is moving, the threshold will be higher than when the robot is standing and waiting for a decision). Typically we set a limit of 1000 candidates to rate. The actual implementation of the rating function takes (initially) 3 ms per candidate (section IV-B gives some timing results for each step of the process), thus in the worst case, it takes up to 3 sec to plan the next view. These steps are depicted in figure 3.

Moreover, the examination process could select the weight of the rating function linear combination depending on the

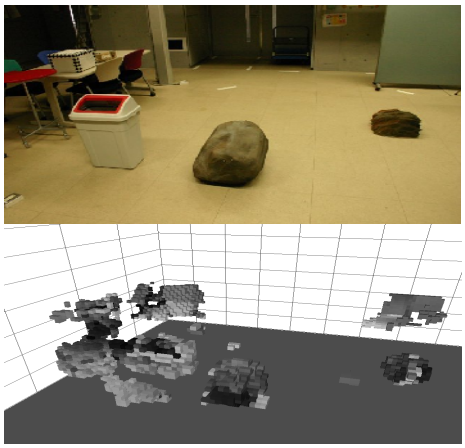

Fig. 4. Real view of the experiment environment and the corresponding 3D occupancy grid generated by the robot

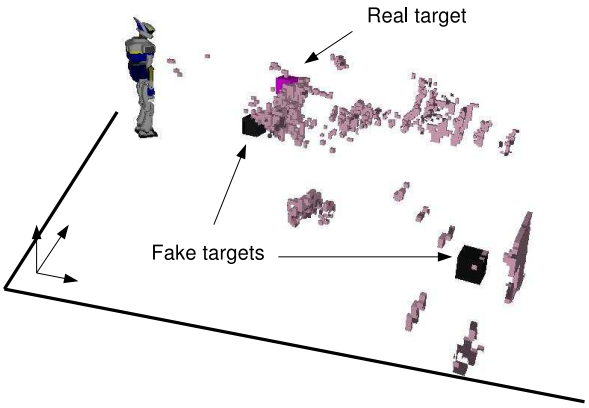

Fig. 6. A screen capture of the simulator at the end of the search behavior

current strategy. When the examination process comes out with a candidate, the existence of a path to the target is then checked using an A^* 2D planner. This simple path planner, takes into account the bounding box of the robot while walking. The planning is done only for the robot body, and the residual head motion is then executed to reach the target sensor configuration.

H. The recognition function & the update process

A simulation of the recognition system has been implemented. Although the simulation is simple, it has the main characteristics of a real recognition system. A random function creates false target that adds some noise in the probability map. Few assumptions are made on the underlaying recognition system and the output of the recognition is a list of object pose with their associated likelihood.

Each object pose is then converted into the corresponding voxel set and their probabilities are merged with the target presence probability map through the update process. The update process will then normalize the distribution probability in order to have: $\sum_{Environment} p(v_i, t) = 1$.

IV. Experiments

A. Object search and exploration behavior

Preliminary experiments were done to validate the algorithm. Two simulations where performed: one in which the target object is not present and another one in which the object is present but not hidden.

In the first experiment, the robot mainly driven by the NI explore the full environment (figure 5). The complete exploration is done in 100 hundred views and lasts 5 minutes (the displacement time is not taken into account). The motion cost weight is very low, thus the system was focusing on retrieving the maximum information at each step whatever displacement it needs ($\lambda_{NI} = 1000$, $\lambda_{MC} = 0.1$, $\alpha_{NF} = 1$, $\alpha_p = \alpha_t = 0$). The table below gives the total distance traveled by the robot for 50 different views, and the remaining unknown voxels in the environment for different values of λ_{MC} ($\lambda_{NI} = 1000$).

λ_{MC}	0.01	0.1	0.2	0.5	2	3
Total distance (m)	91.3	71.4	56.3	45.7	21	16
Unknown (%)	13.8	13.7	13.7	16	21	19

In the second experiment the robot finds the target after 45 views (figure 6). Depending on the settings (the $\lambda_{NI}/\lambda_{DP}$ ratio) the robot will lock the target after the first view or will do some remaining exploration before focusing its attention on the target. An online video[1] shows the complete search sequence.

Next section gives some implementation details and benchmark results on the different parts of the algorithm.

B. Implementation notes

The whole design and implementation were done while targeting a fast and reactive behavior of the robot, thus time constraints are crucial and have guided the project. The table below shows some benchmark results done with a 5cm resolution of a 12x6x4 meter environment, using a 3GHz bi-Xeon workstation with Hyper-Threading™.

Many improvement of the initial code were performed. Concerning the visibility map, the visibility sphere of a point is precomputed according to the R_{min}, R_{max} values and stored in a look-up-table (LUT). Then, the map update is done incrementally, which means that only points which have a change in their state will be considered. Because it is done incrementally, the update process gets faster.

14500 points with no LUT	6s
24600 points with the LUT	3.1s
average for 50 updates with LUT	380msec

The constrain achieved by the visibility map drastically reduces the configurations to consider. The discretized configuration space of the robot sensor in this experiment contains 240x120x200=5.76 million configurations, the visibility map and local maxima extraction only outputs 1000 configurations.

[1] http://staff.aist.go.jp/francois.saidi/video/HRP2SearchBehavior.avi

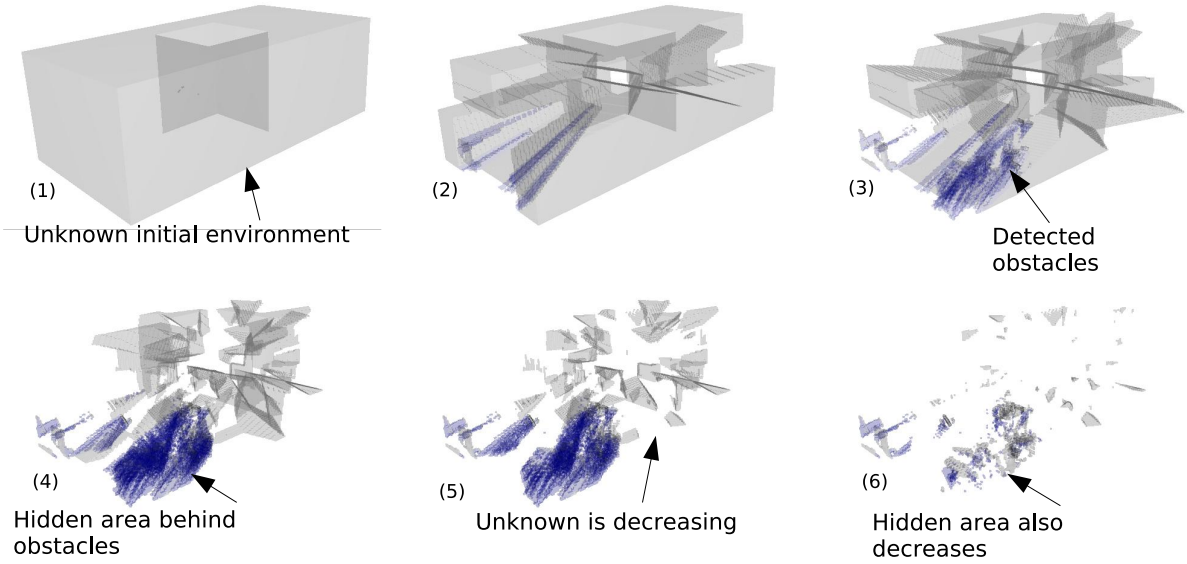

Fig. 5. A screenshots sequence of the exploration behavior performed in simulation

The rating function computation is a highly parallelisable process which benefits of multi-core/cpu machines. Thus, the number of physical/logical cpu is detected at runtime and the corresponding number of threads is used to compute the score of the candidates. Once more, the visibility map update gets faster as the unknown in the environment decreases. Moreover as the environment is being mapped, the number of unknown voxels decreases quickly and the computation of the rating function gets faster. The average computation time over 50 views of the rating function using 4 threads is around 1 msec per candidate.

V. Conclusion

This paper exposed the framework for a search behavior developed for the humanoid robot HRP-2. The problem, which falls in the sensor planning field, is formulated as an optimization problem. The concept of visibility map introduced in [3] to constrain the sensor parameter space according to the detection characteristics of the recognition algorithm is used to reduce the dimension of the sensor parameter space. The rating function uses a formulation of the expected information takes into account a prediction on occlusion in the unexplored space to provide a more accurate information prediction. Simulation results of an exploration and search behavior has been presented to validate the model. Work is on progress, and experiments on the real robot to validate parts of the algorithm are already undertaken and the z axis limitation of the sensor is on the way of beeing removed.

Acknowledgment

This research was partially supported by a Post-doctoral Fellowship of Japan Society for Promotion of Science(JSPS) and JSPS Grand-in-Aid for Scientific Research.

References

[1] Y. Ye and J. K. Tsotsos, "Sensor planning in 3d object search: its formulation and complexity," in *Fourth International Symposium on Artificial Intelligence and Mathematics*, Florida, U.S.A., January 3-5 1996.

[2] ——, "Sensor planning for 3d object search," *Computer Vision and Image Understanding*, vol. 73, no. 2, pp. 145–168, Feb. 1999.

[3] F. Saidi, O. Stasse, and K. Yokoi, "A visual attention framework for a visual search by a humanoid robot," in *IEEE-RAS International Conference on Humanoid Robots*, Genova, Italy, December 4-6 2006, 346-351.

[4] L. E. Wixson, "Gaze selection for visual search," Ph.D. dissertation, Department of Computer Science, Univ. of Rochester, 1994.

[5] V. A. Sujan and S. Dubowsky, "Efficient information-based visual robotic mapping in unstructured environments," *The International Journal of Robotics Research*, vol. 24, no. 4, pp. 275–293, Apr. 2005.

[6] B. O. Koopman, *Search and Screening*. Pergamon Press, 1980.

[7] C. J. Connolly, "The determination of next best views," *IEEE Int. Conf. on Robotics and Automation*, pp. 432–435, 1985.

[8] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, "The 3d linear inverted pendulum mode : A simple modeling of a biped walking pattern generation," in *International Conference on Intelligent Robots and Systems*, Maui, Hawaii, Usa, November 2001, pp. 239–246.

[9] B. Verrelst, K. Yokoi, O. Stasse, H. Arisumi, and B. Vanderborght, "Mobility of humanoid robots: Stepping over large obstacles dynamically," in *International Conference on Mechatronics and Automation*, Luoyang, China, June 25-28 2006, pp. 1072–1079.

[10] O. Stasse, A. Davison, R. Sellaouti, and K. Yokoi, "Real-time 3d slam for humanoid robot considering pattern generator information," in *International Conference on Intelligent Robots and Systems, IROS*, Beijing, China, October 9-15 2006, to appear.

[11] B. Telle, O. Stasse, T. Ueshiba, K. Yokoi, and F. Tomita, "Three characterisations of 3d reconstruction uncertainty with bounded error," in *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, 2004, pp. 3905–3910.

[12] A. Makarenko, S. Williams, F. Bourgault, and H. Durrant-Whyte, "An experiment in integrated exploration," in *IEEE/RSJ International Conference on Intelligent Robots and System*, vol. 1, 2002, pp. 534 – 539.