Cognitive Recognition under Occlusion for a Visually Guided Robotic Errand Service

Ahmed M.Naguib [1], Xi Chen [2], and Sukhan Lee*[3]

School of Information and Communication Engineering, Sungkyunkwan University, Suwon, South Korea
[1]ahmed.m.naguib@gmail.com,[2]pcchenxi@gmail.com,
[3]lsh@ece.skku.ac.kr

Abstract. It is a hard task to develop a reliable vision system for running errands with a service robot in an unstructured indoor environment such as homes. Many visual challenges, such as: perspective, clutter, illumination, and occlusion, need to be handled appropriately. While previous researches addressed these problems from the contexts of either object recognition or object searching, our proposed approach relies on a solution that combines these two as one. We are proposing a "Cognitive Recognition" System, where information gathered from scene recognition helps deciding the next optimal perspective, and environmental parameters measurements determine the uncertainty in recognition measurements and thus the proper probability map update used in object search. We show particularly in this paper how this approach provides a practical solution to clutter and occlusion environment. And we demonstrate the results with our HomeMate Service Robot.

Keywords: Object Recognition; Evidence Collection; Object Searching; Occlusion Management; Cluttered Environment; Illumination Variation; Visually-Guided Service Robot; Errand Service for Elderly

1 INTRODUCTION

Since service robot environment is highly dynamic and unpredictable, researches on vision systems for service robots are relatively new compared to Industrial robots systems. There are many challenges to overcome for a practical service robot to function in human environment [1], such as: Human Present, Dynamic noisy and clutter environment, Perception, Obstacle-aware navigation, and many others.

Vision challenges, in particular, were studied extensively within last decade and many solutions were proposed to overcome most of the issues. These solutions, however, approached the problem from two different views: Object Recognition, and Object Searching. Object Recognition approach tries to find the object under any condition to be expected in a scene, while Object Searching tries to find optimal perspective of sensors given a certain environment.

From Object Recognition perspective, previous researches can be categorized into: 1) single feature recognition systems, such as: geometric-based features BOR3D [2], photometric features MOPED [3] and SIFT [4]. 2) Multi-features recognition systems

adfa, p. 1, 2011.

such as: simple fusion of 3D lines, SIFT and color [5], [6], features combine using probabilistic evidence-based reasoning [7], and using Bayesian framework [8]. To overcome environment variations, redundancy provided by multiple features, as well as an adaptive probabilistic fusion framework is a necessity.

Similarly in searching, Ye and Tsotsos [9] first formulated object search as an active vision problem, where an efficient trajectory of camera views, that localizes the target object, is sought. Garvey [10] proposed the idea of indirect target search. Wixson et al. [11] elaborated the indirect search idea and have shown efficiency gains both theoretically and empirically. [12] And [13] Used spatial relations between objects to search more efficiently, and locate good views of the target object. Kollar and Roy [14] showed that object-object and object-location co-occurrence statistics can be used to predict

As mentioned before, most of the researches, however, treat recognition and searching as two independent parts. The general flow of an object searching task is: first the searching system searches the space and drives the robot to the place where camera can see the target object; as soon as the target appears in camera view the recognition system will quickly identify the target and the task is over. There are several unrealistic assumptions inside this process. For example, camera perspective may cause recognition system to fail in recognizing an object. Occlusion, Distance, intensity, and severe clutter may exist in a perspective image and stress recognition system. The object itself may not be placed directly in front of robot camera. The current researches do not give a sufficient solution on what to do next after a recognition fails.

We have developed two novel frameworks for an adaptive Bayesian recognition system with multiple features, as well as an active object searching and evidence collection system. We also have integrated the two frameworks into a “Cognitive Recognition” system that drives a service robot to locate a target object in an indoor environment.

The contribution of our proposed method is the information sharing approach between recognition and searching system to be a complete cognitive system. By using a geometric-based feature (3D shape descriptor), recognition system can share the observed environment information with searching system. With a clear understanding of the environment, searching system not only drives the robot to poses where target object is likely to exist, but also controls the distance and angle to better see the target. Using our proposed method the best view with maximum probability of finding the target object can be calculated accurately and problems such as occlusion and object orientation can also be solved easily.

In this paper, we will present the capabilities of our system to overcome typical severe conditions, such as occlusion. Finally, we demonstrate our method using a mobile robot. This system is being currently used in our 3rd generation service robot, Korus HomeMate as shown in [15], which is equipped with an MS Kinect RGBD sensor, Bumblebee 2 stereo camera, as well as an onboard Intel Core i7 notebook. This robot has been used in several elderly-centers as a test and evaluation prototype.

In section 2, a brief introduction of Recognition system is presented with detailed description of segmentation, 3D shape descriptor, and occlusion treatment process; a general description about object searching is presented in section 3; lastly, experimental results and conclusion are provided in section 4 and 5 respectively.

2 ADAPTIVE BAYESIAN RECOGNITION FRAMEWORK

We are proposing an Adaptive recognition system that can achieve reliable results under very hard and unpredictable environmental conditions. Proposed recognition system consists of: primitive surfaces extraction and definition of volumes of interest, optimized rapid voxel representation of 3D environment, simultaneous multiple independent feature extraction, classification and matching, Evidence-based Bayesian structure for feature fusion, and a high-level adaptive component for adjusting system parameters according to environmental conditions.

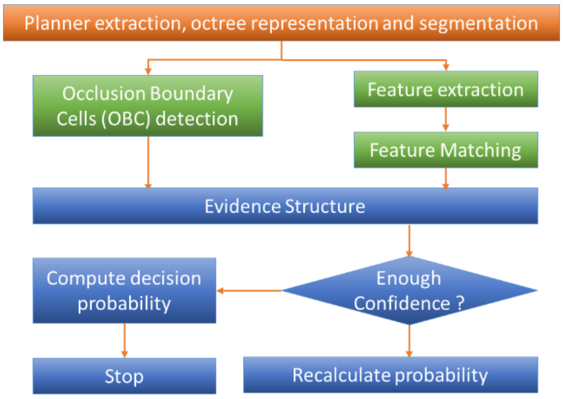
Planner extraction, octree representation and segmentation
Occlusion Boundary Cells (OBC) detection
Feature extraction
Feature Matching
Evidence Structure
Compute decision probability
Enough Confidence ?
Stop
Recalculate probability

Fig. 1. Flow diagram of the recognition framework

To follow the scope of this paper, we will briefly introduce rapid voxel representation, and features, then we will express Occlusion Boundary Detection and Bayesian classification algorithm. Fig. 1 is the flow diagram of the recognition framework. Further detailed explanation and evaluation of proposed classification and recognition system is presented in [16]

2.1 Octree Representation, Segmentation and Planner Extraction

Since most of geometric based feature extraction and processing require frequent access to geometric bit neighbors, having a representation that can provide an instant access to neighbors is the key stone for a real-time geometric analysis. Dealing with 3D sensor point cloud poses many issues. Not only it has a high 3D neighbor determination cost, but also it usually has high density points as well as having depth error characteristics and amount of point that are directly dependable on used sensor.

Since, an indoor service robot is only interested in a limited volume for searching, we use a RANSAC-based planner extraction for detection of contexts of interests. We then generate octree representation for these volumes of interest and perform a rapid displacement-based segmentation to locate candidate objects

2.2 Extracted Features

Measuring multiple features provide redundancy that overcomes many issues due to environmental variations. For example, when illumination is not proper for a photometric feature, geometric features may still provide sufficient evidence for decision. The opposite is true in case of highly cluttered environment. In our proposed system, we rely on simple nine dimensional shape descriptor, SIFT, and seven dimensional appearance vector per color (Table 1).

Table 1. Selected Features

Feature Type	*Feature Name*	*Description*
Geometric Features	Object Height	Distance from top cell of object representation to context surface
	Top Area	Top layer convex hull surface area
	Circle Surface	Ratio of number of cells outside circler boundary to the total number of visible cells
	Top Shape	2D vector consists of top layer width and top layer length
	Body Shape	2D vector consists of top layer width and middle layer width
	Body Mass	2D vector consists of object volume and concave index of top layer
Photometric features	SIFT	Scale-invariant feature transform
	Mean Hue	Average Hue of a color appearance vector
	Mean Saturation	Average Saturation of a color appearance vector
	Scale	Largest eigen value of a color appearance vector in HS Space
	Orientation	Angle of largest eigen value of a color appearance vector in HS Space
	Photometric Ratio	Ratio of eigen values of a color appearance vector in HS Space
	Pixels Percentage	Percentage of pixels of a color appearance vector relative to object pixels
	Scattering Index	Geometric scattering Index of pixels of a color appearance vector

2.3 Bayesian Classification for Feature Fusion

Each feature provides a likelihood measurement for a candidate object to be target object. In order to fuse them and compute final probability of that candidate object being the target object, we use Bayesian reasoning as follows:

$$P(c = o_i \mid M^c) = \frac{P(M^c \mid c = o_i)P(c = o_i)}{P(M^c)}$$

$$= \frac{P(M^c \mid c = o_i)P(c = o_i)}{P(M^c \mid c = o_i)P(c = o_i) + P(M^c \mid c \neq o_i)P(c \neq o_i)}$$

$$= \frac{1}{1 + \frac{P(M^c \mid c \neq o_i)P(c \neq o_i)}{P(M^c \mid c = o_i)P(c = o_i)}} \quad (1)$$

$$P(M^c \mid c \neq o_i) = \text{argmax}_{j\neq i}\{P(M^c \mid c = o_j)\}$$

Where o_i is target object, c is candidate object, M^c is measurement of candidate object, $P(M^{c_i} \mid s_i = o_i)$ is positive likelihood, $P(M^{c_i} \mid s_i \neq o_i)$ is negative likelihood, and $P(c = o_i)$ and $P(c \neq o_i)$ are priori information. For first iteration, priori probabilities are missing, we assume they are equi-probable. For following iterations, we carry out priori information from each previous iteration. Finally, we assume for practical point of view that each feature is independent from the others, thus, their mutual likelihood is merely the production of their individual likelihoods

2.4 Occlusion Boundary Detector and Probability Fusion

Since many of 3D Shape descriptor features are very sensitive to specific spatial parts of the measured object, even minor occlusion of these parts can affect dramatically matching results. To control this uncertainty, we are introducing an algorithm for detecting occlusion boundaries (Fig. 2) of each segment and a confidence management system for each geometric feature measured from the scene.

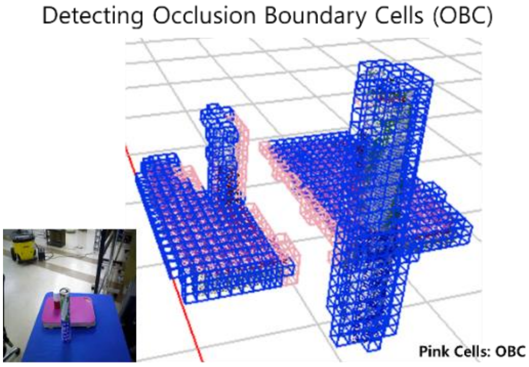

Fig. 2. Example of detected Occlusion Boundary Cells (OBC)

Since every point in the scene is measured with reference coordinate frame of the 3D sensor, ray tracing octree cells is relatively an easy task to do. We first locate cells that are on similar line of sight with the camera and group them. For each group, we measure distance between the foreground cell and background cell (d2), and shortest distance between foreground cell and 3D line connecting background cell and camera center (d1) (Fig. 3)

We call background cell as a boundary cell if the following constraints are met:

$$\text{d1} \leq \sqrt{2}\, x\ cell\ size \quad AND \quad \text{d2} \geq 2\, x\ cell\ size$$

After detecting occlusion boundary cells (OBC) of segmentation, we localize occlusion area in an object using a simple vector orientation and update the likelihood probability accordingly (Equation 1). The more an occlusion affects a certain feature, the more this feature likelihood will approach 50%. At complete uncertainty, likelihood reaches en equi-probable point. We use a linear approximation to scale likelihoods appropriately.

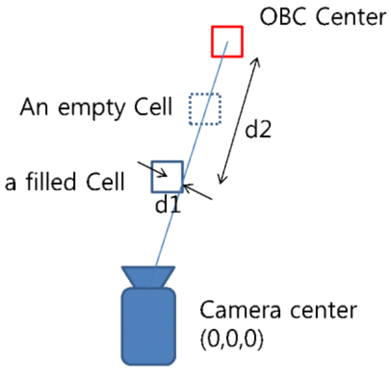

Fig. 3. Occlusion Boundary Cell (OBC) detection constraints

3 ACTIVE EVIDENCE COLLECTION

As mentioned in introduction, there are lots of factors that may cause recognition system to fail. We list in Table II some typical challenges that can greatly affect the performance of the recognition system.

We have introduced several novel approaches to properly treat these challenges inside recognition system. However, by taking advantage of the mobility of the robot, our object searching algorithm is designed such that it collects evidences and information from recognition system and environment so that it avoid these challenges especially occlusion as much as possible in the next robot pose / camera perspective. This redundant in the system provides reliable solution and is the foundation of our cognitive object recognition, especially for challenges that cannot be solved unless robot is moved to a better view of observation, such as the robot is looking at the surface of a target object that has very no features or it is total occluded. Instead of making decision with less evidence at the first time it will be much wiser to move to a better view first and process recognition under a reliable condition. Two stages will be described in the active evidence collection: basic concepts of searching space and action space; the calculation of the next best pose and the obstacle detection and avoidance.

Table 2. Typical Problems in Recognition

Factor Type	***Factor Name***	***Description***
Environment	Illumination	Environment illumination
View of observation	Distance to camera	Distance form object to camera
	Object orientation	The surface that camera is looking at
	Occlusion	Partially or total occluded by other object

3.1 Searching Space and Action Space

As descripted in [17], a searching region $\mathbf{\Omega}$ is a 3D space with known boundaries and context inside. The region $\mathbf{\Omega}$ is tessellated into a 3D grid of non-overlapping cubic elements c_i, $\mathrm{i} = 1 \dots \mathrm{n}$. A action space $\mathbf{\Psi}$ is the set of cells we can move the robot on and from where the robot can detect at least one cell in $\mathbf{\Omega}$. By using spatial relation and maximum/minimum detectable distance, $\mathbf{\Omega}$ is reduced to be the area of the contexts which related to the target object and $\mathbf{\Psi}$ is reduced to be the area that around $\mathbf{\Omega}$ within the detectable distance. Fig. 4 give an example of searching and action space.

3.2 The Next Best Pose

We define $\mathrm{s} = \mathrm{s}(c_i, c_j, \mathrm{a})$ as action for each pose, where $c_i \in \mathbf{\Psi}, c_j \in \mathbf{\Omega}$ and a is the algorithm used for recognition. By standing at cell c_i in action space $\mathbf{\Psi}$ and looking at c_j in searching space $\mathbf{\Omega}$, the utility of the action s – the probability of detecting the target object by apply action s is defined as:

$$\mathrm{u(s)} = \sum_{i=1}^{n} \mathrm{p}(c_i) * d(c_i, s)$$

where n is the number of cells in searching area that inside the recognizable range of action s, $\mathrm{p}(c_i)$ is the probability that the target object is located at c_i and $d(c_i, s)$ is the probability of detecting the target object at c_i by applying action s. if c_i is outside of the field of view of action s, if c_i is outside of the field of view of action s, $d(c_i, s) = 0$; $d(c_i, s) = 0$; if c_i is inside the field of view and is not occluded by any other object, $d(c_i, s)$ is a function of recognition algorithm, distance from c_i to the camera, object orientation and the environment illumination. Detailed description is in [16]

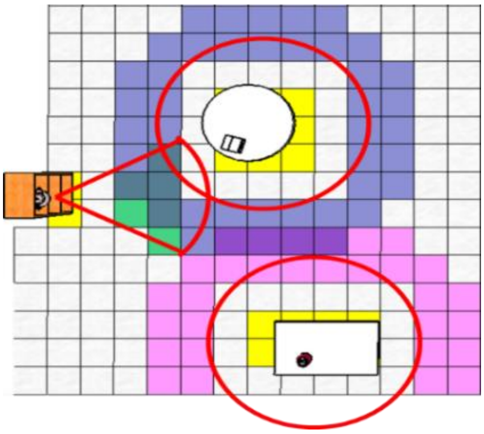

Fig. 4. One example of search space Ω and candidate space Ψ. The cells in yellow belongs to Ω and the cells in other colors belongs to Ψ. The red line represents the accessible direction of context, and the circle means 360 degree accessibility.

The value of pose is not only depends on the number of cells in recognizable range but also the value of detectability probability $d(c_i, s)$. The distance, object orientation and the environment illumination is easy to measure and calculate but the status of occlusion is more difficult to handle. An actuate detection of occlusion can help us to make reasonable judgment of the current environment, a better calculation of pose utility and avoid the obstacles for the next observation.

3.3 The Occlusion Detection and Avoidance

After each recognition process, the objects in the current scene are segmented and send to the searching system with their height, weight and breadth and the center location. We first project the point cloud of the table on the table 2D surface and register the object we received from the recognition on the map, then the areas which are still empty is occluded area in the current scene. Fig. 5 shows an example of object registration and occluded area detection.

We already proposed a method in the recognition system of how to detect occlusion. In the recognition system we only need to apply one time on the current scene, but here we need to calculate the occlusion for every cells in the action space. If we want to use the same method we need to firstly transform all information of cells and objects into the current camera view. When the size of action space increase the calculation of transformation may cause serous system delay. Furthermore, here we do not need as high accuracy as what we did in the recognition framework. Instead of using the previous method we developed a simply and fast method of roughly detection occlusion.

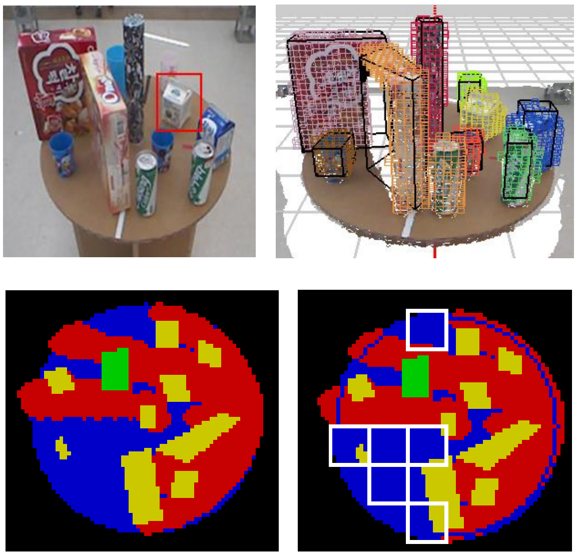

Fig. 5. An example of recognition output. Top left image taken by camera, the target object is marked as red. Top right the result of object segmentation. An example of occlusion detection. Bottom left object register and table point clout projection. Bottom right result of occlusion cell detection

We call a cell or an object visible if it satisfies two conditions: it is inside current camera Filed of View (FOV) and it is not occluded by other object. We define a camera configuration as $(x_c, y_c, z_c, \mathrm{p}, \mathrm{t}, \mathrm{f}, V_V, V_H)$ where V_V and V_H are the camera limitation of vertical and horizontal view. For a cell or an object which is centered at point $\mathrm{p} = (x_p, y_p, z_p)$, we calculate the horizontal and vertical angle component (v_v, v_h) of vector p in camera coordinate. If $v_v < \frac{V_V}{2}$ and $v_h < \frac{V_H}{2}$ we say the cell or object centered at point p is in FOV.

We represent an object $\boldsymbol{O}$ as a box with the same height, weight and breadth as the segmented object. A cell or object centered at point $\mathrm{p} = (x_p, y_p, z_p)$ can only be occluded by object $\boldsymbol{O}$ within a fixed range. This range is bounded by the top, down, left-most and right most corner points of $\boldsymbol{O}$. If camera is located inside this range, where p is occluded, we say the cell or the object centered at p is occluded. Several example of occlusion detection based on camera position is shown in Fig. 6. The occluded angle range for a cell or an object is then independent with camera position. The occluded angle range only need to be calculated one time and for each camera position we only need to check if it is inside any of these occluded angle range.

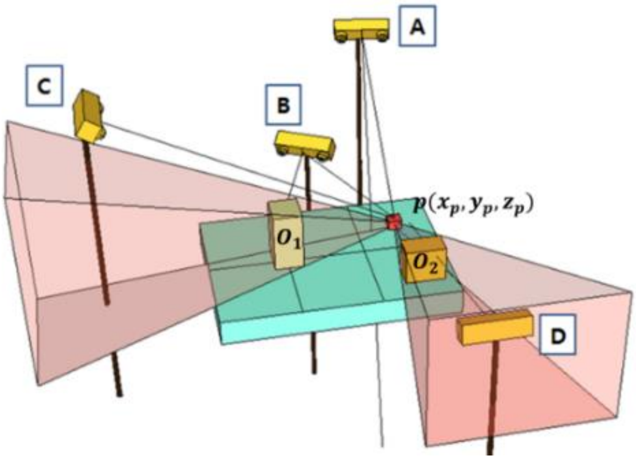

Fig. 6. O_1 and O_2 are two objects on a table, and point $\mathrm{p} = (x_p, y_p, z_p)$ is the point to be checked. A, B, C, D are 4 camera positions. Point p will be occluded in the red area. For the 4 camera A, B, C, D point p is visible from A; p is outside FOV from B; p is visible from C; p is occluded by O_2 from D

4 EXPERIMENT RESULT

We tested our algorithm under various illumination, occlusion and cluttering conditions. Statistical results are shown in Fig. 10. In the experiment shown here, we demonstrate the ability of our proposed algorithm to search and locate object occluded in a highly-cluttered environment. Other experiments are available at [15].

First, we placed one tables, with highly cluttered objects, in the test environment, as shown in Fig. 7. The size of the searching environment was 2.2m × 3.6m (width x length) and the size of each table was 0.6m × 0.6m (width x length). We, then, placed a target object, the “yellow cup”, between two large obstacles. Due to the severe occlusion casted by obstacles, only accessible angle can see the target object, which makes this setting very challenging. Finally, we directed our mobile robot to search and locate the target object using the proposed algorithm.

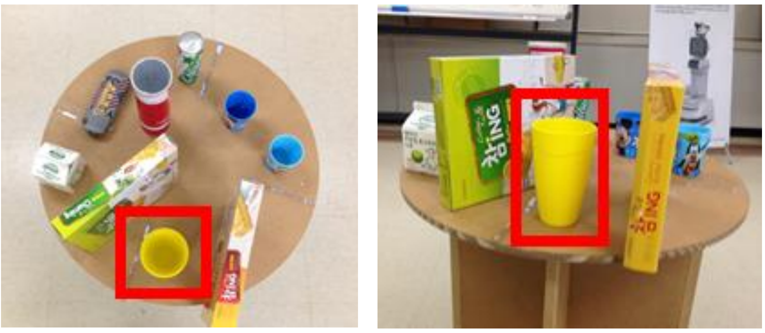

Fig. 7. Objects on the table. The target is in between two obstacles

The first pose was calculated using the proposed algorithm and the robot move to the selected location. The image taken by camera and the recognition result is shown as Fig. 8. The recognition system extract environment and share it with searching system. Knowing the position and size of objects on the table occluded area can be defined. After probability updating, searching algorithm generate the second pose to see the occluded area, as shown in Fig. 9. The target object became visible and the recognition system was able to locate it successfully.

Since there are many parameters that affect the performance of the proposed system, such as the amount of obstacles in the environment, the location of the target object, and the initial location of the robot, etc, we presented the above case study to demonstrate the performance of the system. Furthermore, additional experiments were conducted and can be found in [15]. Additionally, for every object of 7 possible target objects, we ran the robot with the proposed system three times under each of four distinct environmental conditions. We recorded the maximum number of poses the robot took to locate the target object as shown in Fig. 10 (in this particular experiment, we used one context placed in the middle of the 2.2m x 3.6m map, and robot initial position is fixed at the corner).

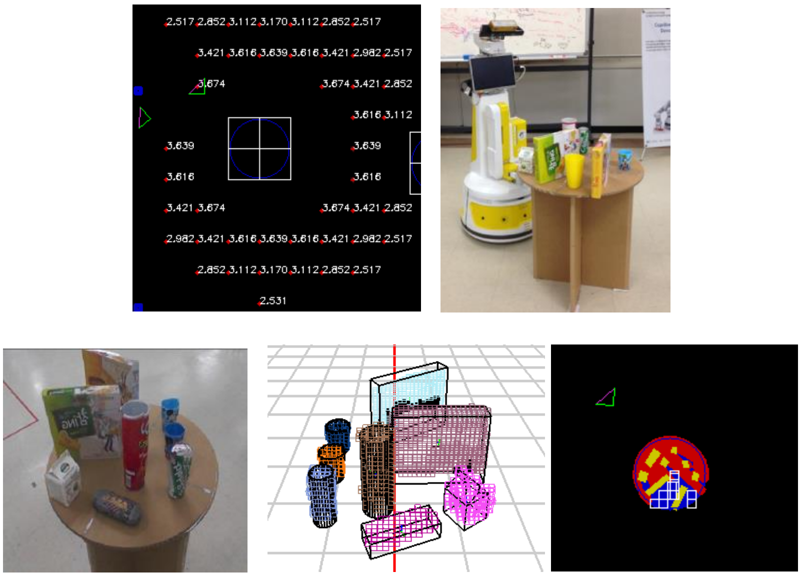

Fig. 8. First pose to access the table (top). RGB image taken by camera (bottom left); View point of recognition result (bottom middle); occluded area detection (bottom right).

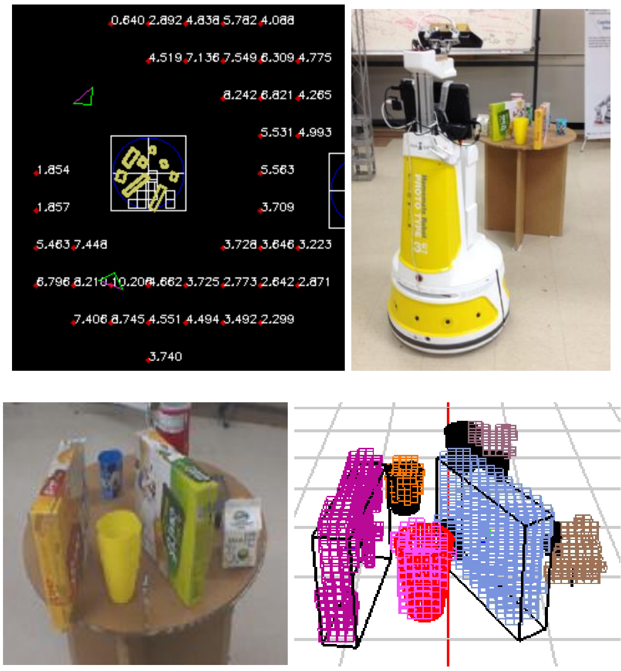

Fig. 9. Second best pose that covers occluded cells (top). RGB image taken by camera (bottom left); View point of recognition result and the target is marked as red. (bottom right)

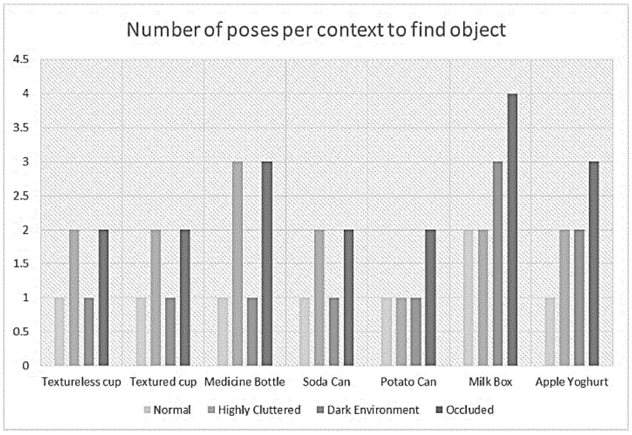

Fig. 10. Performance of Cognitive Recognition system under various environmental conditions

5 Conclusion

In this paper, we have introduced our Cognitive Recognition System. We have explained both Bayesian adaptive recognition, and Active evidence collection components. We have shown how overall system can benefits greatly from knowing environmental conditions as well as information of surrounding objects. We have shown how both components can detect occlusion and take proper action to avoid it. Experimental results show that this framework allows the object searching and recognition system to detect/avoid occlusion, manage measurement certainty and find the optimal perspective to see occluded object.

6 References

1. Kemp, C.C., Edsinger, A., Torres-Jara, E., "Challenges for robot manipulation in human environments [Grand Challenges of Robotics]," March 2007 Robotics & Automation Magazine, IEEE (Volume:14 , Issue: 1)
2. M. Bertsche, T. Fromm, and W. Ertel, "BOR3D: A Use-Case-Oriented Software Framework for 3-D Object Recognition," 2012 IEEE Conference on Technologies for Practical Robot Applications, Woburn.
3. A. Collet, M. Martinez, and S. S. Srinivasa, "The MOPED framework: Object Recognition and Pose Estimation for Manipulation," Sep. 2011 the International Journal of Robotics Research, vol. 30, no. 10, pp. 1284–1306.
4. F. A. Pavel, Z. Wang, and D. D. Feng, "Reliable Object Recognition using SIFT Features," MMSP'09, October 5-7, 2009, Rio de Janeiro, Brazil
5. S. Lee, S. Lee, J. Lee, D. Moon, E. Kim, and J. Seo , "Robust Recognition and Pose Estimation of 3D Objects Based on Evidence Fusion in a Sequence of Images," 10-14 April 2007 IEEE International Conference on Robotics and Automation Roma, Italy.

6. S. Lee, E. Kim, and Y. Park , “3D Object Recognition using Multiple Features for Robotic Manipulation,” May 2006 IEEE International Conference on Robotics and Automation, Orlando, Florida
7. S. Lee, Z. Lu, and H. Kim, "Probabilistic 3D Object Recognition with Both Positive and Negative Evidences," 2011 IEEE International Conference on Computer Vision.
8. H. Kim, J. Lee and S. Lee, “Environment Adaptive 3D Object Recognition and Pose Estimation by Cognitive Perception Engine,” 2009 IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA).
9. Y. Ye, J.K. Tsotsos, “Sensor planning for 3d object search, Comput. Vis. Image Understand,” 73 (2) (1999) 145–168.
10. T.D. Garvey, “Perceptual strategies for purposive vision, Technical report, SRI International,” 117, 1976.
11. L. Wixson, D. Ballard, “Using intermediate object to improve efficiency of visual search,” Int. J. Comput. Vis. 18 (3) (1994) 209–230.
12. K. Sj¨o¨o, D. G´alvez-L´opez, C. Paul, P. Jensfelt, and D. Kragic, “Object search and localization for an indoor mobile robot,” J. Computing and IT, vol. 17, no. 1, pp. 67–80, 2009.
13. A. Aydemir, K. Sj¨o¨o, J. Folkesson, A. Pronobis, and P. Jensfelt, “Search in the real world: Active visual object search based on spatial relations,” in ICRA, 2011.
14. T. Kollar and N. Roy, “Utilizing object-object and object-scene context when planning to find things,” in ICRA, 2009.
15. HomeMate errand service demo: http://www.youtube.com/watch?v=4ENFcHP1GaI
16. Ahmed M.Naguib, Sukhan Lee, “Adaptive Bayesian Recognition with Multiple Evidences,” 2014 The 4th International Conference on Multimedia Computing and Systems (ICMCS) (To be published by April 2014).
17. Xi Chen, Sukhan Lee, “Visual search of an object in cluttered environments for robotic errand service,” 2013 IEEE International Conference on Systems, Man, and Cybernetics (SMC).