Design a Human-Robot Interaction Framework to Detect Household Objects

Sadi Rafsan[1], Safayet Arefin[1], A. H. M. Mirza Rashedul Hasan[2], and Mohammed Moshiul Hoque[1]
[1]Chittagong University of Engineering & Technology, Chittagong, Bangladesh
[2]Rajshahi University, Rajshahi, Bangladesh
e-mail: sadi.rafsan@gmail.com, safayet_arefin@yahoo.com, mirza_iu@yahoo.com, mmoshiulh@gmail.com

Abstract— **In human-robot interaction scenarios, the ability to identify a single object from multiple objects is an important task for service robots. Although there has been recent progress in this area, it remains difficult for autonomous vision systems to recognize objects in natural conditions. The service robot should detect a particular object according to the user's demand. This paper describes a human robot interaction framework to detect a particular household object from multiple objects through text-based interaction. Haar Cascade Classifiers is used to detect objects and developed a user friendly interface for human-system interaction. The propose framework use color, size, or position information to distinguish the user requested object in multi object scenarios. Evaluation results shows that the system is quite effective to detect the target household object from multiple objects in real time.**

Keywords-Human-robot interaction; interactive object detetction; object recognition; classification; evaluation

I. INTRODUCTION

Due to recent advances in technology and computing, it is now possible to use assistive technology or helper robots for a much wider range of tasks than ever before. It is human's dream to let the robots take on tedious, boring or dangerous work so that they can commit their time to more creative tasks. Nowadays, there have been developed many interesting robots. But unfortunately, the intelligent part seems to be still lagging behind. In order to work with the real environment independently, the robot should be capable of performing lots of social and cognitive activities in socially acceptable manner. Although there are lots of research challenges remain unsolved related to the autonomous robots, in this paper wwe would like to focus one of the crucial issues such as household object detection and recognition by robots.

Household object detection deals with identifying the presence of various household objects in real-time in various lighting conditions, and with various backgrounds. The visual appearance of objects can change enormously due to different viewpoints, occlusions, illumination variations or noise. Furthermore, objects are not presented alone to the vision system, but they are immersed in an environment with other elements, which clutter the scene and make recognition more complicated. Humans recognize a multitude of objects in images with little effort, despite the fact that the image of the objects may vary somewhat in different view-points in many different sizes and scales or even when they are translated or rotated. Objects can even be recognized when they are partially obstructed from view. This task is still a challenge for computer vision systems. Many approaches to the task have been implemented over multiple decades but no one is completely efficient. Here, we proposed a framework that can detect target object from multiple household objects using human-robot interaction scenarios.

In this work, we proposed a framework that can detect a specific household object requested by the user. Here, our concentration is on generating dialogue. Helper robot chooses an appropriate method based on user instruction. The robot receives instructions through the user's speech or text and according to user feedback, it detects the particular object. Let consider a scenario as in Fig.1, there are multiple objects on the table: two glasses, one can and one bottle. From these, user asks for a glass. But there are two glasses which are different in color. So, robot informs it to user and by exchanging some short simple conversation it will finds the target object. Object detection is relatively simpler if the machine is looking for detecting one particular object than recognizing all the objects because it requires the skill to differentiate one object from the other, though they may be of same type. Such problem is very difficult for machines, if they do not know about the various possibilities of objects.

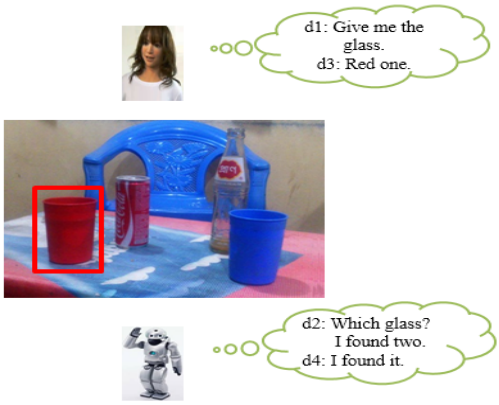


Fig. 1. A HRI scenario of detecting target object

II. RELATED WORK

A recent trend in human detection is to combine multiple information sources, e.g., color, local texture, edge, motion, etc. [1]. In [2, 3], sparse learning techniques is used to select a small number of features and construct cascade classifiers. Hough forests, which performs a generalized Hough transform for object detection, was proposed in [4] and achieved high detection accuracy in benchmark detection datasets. Viola and

Jones [5] proposed the method for object recognition which constructs a cascade of simple classifiers using a learning algorithm based on AdaBoost. There has been a lot of research on robot systems understanding the scene or their tasks through interaction with the user [6, 7, 8, 9]. The robot makes quires to the user to understand the target objects that the user has in mind. Yamakata et al. [10] have presented a probabilistic reasoning method based on a belief network of the object reference. Mansur et al. [11, 12], proposed an interactive method where a human user may be asked to instruct the robot to describe the target object mainly by its color. They developed a vision system to detect objects requested by a user through simple expressions. This research reveals the importance of connecting 'symbolic expressions' with the 'real world' in human-robot interaction.

Though there have been many works, accurate detection is still a major interest in household object detection. Among these works most of them are very complex to implement and are not user friendly. Some of these are not implemented in real time or costly. In this work, we propose HRI framework to recognition the household objects in real time.

III. Preliminary Experiment

We first examined how humans describe an object in multi-object scenarios. We found two main ways. One is to describe attributes of the object, such as color and shape. The other is to mention the spatial relationships of the object in relation to other objects. Based on these results, we propose that an interactive vision system should be able to understand such text-based expressions used by humans.

A total of 10 participants participated in this experiment. The average ages of participants are 24.4 years. Most of them are students of Chittagong University of Engineering and Technology and some are job holders. We have examined how humans describe objects. Participants were asked to choose a preference when multiple objects of same type found. Before the experiment, we explained to the participants that the purpose of experiment was to evaluate the suitable choice for detecting the specific target object. Each trial started with showing the participants a formation of objects. A scene of the experiment is shown in Fig. 2. We have told the participants to provide the following evaluation in each trial.

Fig. 2. Participant describing object in different scenario

A. Case-1: Objects of different size

In this case, we have showed the participant objects of different size like the scenario shown in Fig. 3 (a) and found that maximum of them chose "size" like as small, big, medium to get a specific object. Few choose "position" like middle one and some chose color of largest area in describing the object.

(a) (b)

Fig. 3. (a) Objects of different size (b) Objects are of same size but different in color

B. Case-2: Objects of same size but different in color

In Fig. 3(b) shown the case when objects have same size but are different in color, maximum participants choose "color" like as green, blue etc. to get a specific object. Few choose position in describing the object.

C. Case-3: Objects of different or same size and color

When objects have different or same size and color [Fig. 4], most of the participants choose "position" like as left one, 3rd from left etc. to get the desired object. Some of them used both color and size in describing the object such as, big green one etc. Few used both color and position such as, red one: 3rd from left.

Fig. 4. Objects of Different or Same Size and Color

D. Data Analysis

After considering the three cases for human behavior analysis we have found that from 70% of the participants choose "size" in case 1, 90% choose "color" in case 2 and 70% choose "position" in case 3 in order to get the specific object [Fig. 5]. From this analysis, we can say that in real environment humans describe objects differently in different situations.

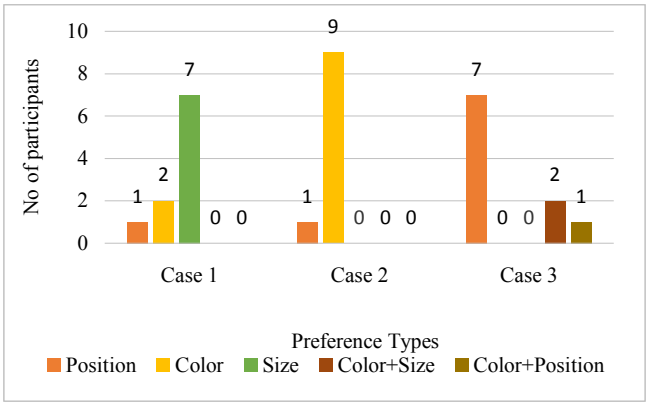


Fig. 5. Data analysis of human behavior in different cases

IV. Interactive Object Detection Framework

The main objective of our work is to design a human-robot interaction framework that can detect a specific household object requested by the user from multiple objects. The proposed framework is illustrated in Fig. 6.

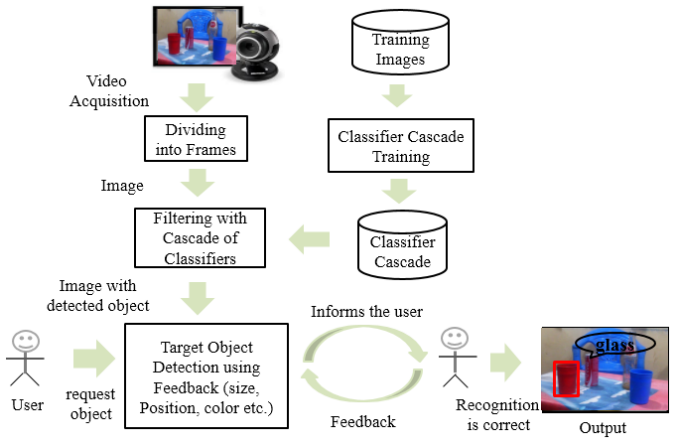


Fig. 6. Interactive household object detection and recognition method

A. Video acquisition

Video acquisition mainly involves obtaining the live video feed of the environment where the target household object is placed. Video acquisition is achieved by making use of a webcam. Taking live video as input, then it needs to convert into a series of frames which are then processed. Our system runs at around 15 frames per second working only with the information present in a single gray scale image to achieve higher frame rates.

B. Filtering with cascade classifier

Frame grabbed from last step need to be converted into gray scale image for sliding window segmentation. Then we need to normalize brightness and increase contrast of the image frame. In each and every frame, then a scan goes on which tries to detect the household object class. This is achieved by making use of a set of pre-trained Haar-cascade classifier. The function used here, finds rectangular regions in the given image that are likely to contain objects the cascade has been trained for and returns those regions as a sequence of rectangles. it scans the image several times at different scales. Each time it considers overlapping regions in the image. It may also apply some heuristics to reduce number of analyzed regions, such as Canny pruning. After it has proceeded and collected the candidate rectangles (regions that passed the classifier cascade), it groups them and returns a sequence of average rectangles for each large enough group. Drawing the rectangle regions, finally we get the image with detected object. Fig.7 shows the detected object after filtering with cascade classifier.

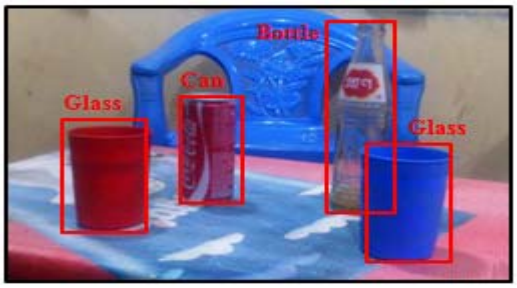


Fig. 7. Detected objects after filtering with cascade classifier.

V. Target Object Detection using Human-Robot Interaction

When multiple objects found, this module detect the specific household object requested by the user, using some short simple user friendly conversations. Let consider a scenario as shown in Fig. 9 where multiple objects are found on the table. User wants the smallest one from three bottles. In this scenario, this module detects the target object using some simple conversations to communicate with the user such as:

User: Give me a glass.
System: I got two glasses. Which one?
User: Red one.
System: I got it.

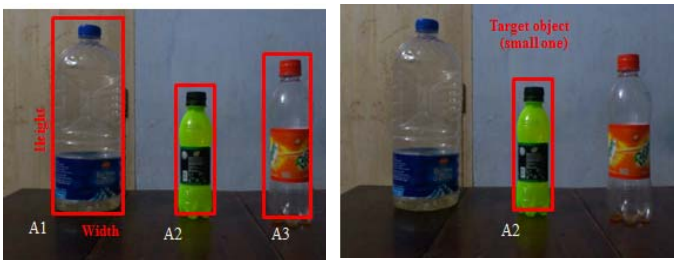
Fig. 9. Target object detection using human-robot interaction

According to user demand, the system detects the target object by comparing size, position, and color information of detected regions. These can be classified into three sub-module: Detection by Size, Detection by Color and Detection by Position

A. Detection by size

When user asks for a specific object by describing its size, the system calculates the area of every region of the detected objects by using the formula in (1).

```
Area = region[i].Width * region[i].Height;(1)
```



(a) Detected all objects (b) Detected requested object

Fig. 10. Target object detection by size.

Calculated the area according to (1), we get: A1=7100, A2=2860, A3=4274. After computing the area it compares all the detected regions, and according to user request in the scenario, it detects the particular small bottle [Fig. 10 (b)].

B. Detection by color

When detecting by color, it converts the detected region of all objects into HSV color space and compare the hue, saturation and value with respective components of desired color. Matching nearest color, system detects the target object according to user feedback. We proposed few rules for color matching as represented in Table I.

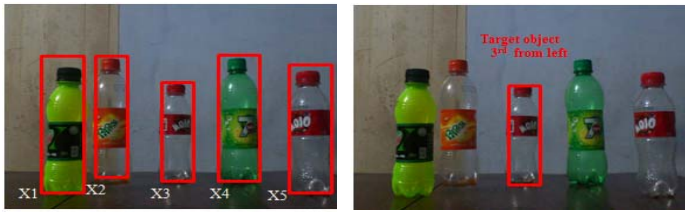
TABLE I: RULES FOR COLOR MATCHING

Color Choice	Condition Hue(H), Saturation(S),Value(V)
Red	14>H>165&& S>53
Green	34<H<90&& S>53
Blue	102<H<160&& S>53
Orange	13<H<25&& S>53
Yellow	25<H<45&& S>53
White	S<27&& V>190
Black	V<25

According to color choice system checks the condition and output the region for which it matches. When it found no matches, it calculates the difference of hue component between color choice and detected region's color (average) and output the region of minimum difference. Thus the system detects the specific object described by color.

C. Detection by position

When user asks for a specific object describing position, it finds the positions of all region comparing, `region[i].X with region[i+1].X`, where, `X` is x-co-ordinate of object region. Fig. 11 shows an example of target object detection by position. Here, X-coordinates of each object region are: X1=80, X2=180, X3=270, X4=350, X5=420.



(a) Detected all objects (b) Detected requested object

Fig. 11: Target object detection by position

According to user feedback, it detects the third bottle sorting the position. Finally, it shows the output by drawing rectangle around the particular household object, as well as, it shows its name. Thus the whole system works to detect specific household object using human-robot interaction.

VI. TRAINING

In training phase, we need to train the system to learn which is object and which is not. Here, after Haar cascade training, it generates our own cascade classifier for Haar features. For training to create Haar-like Classifier we need to follow some steps.

A. Collecting image database

To train cascade classifier at first, we need to collect a large set of positive and negative images. The positive images are those images that contain the object (e.g. glass, bottle etc.), and negatives are those ones which do not contain the object. Having more number of positive and negative (background) images will normally cause a more accurate classifier. The images need to be converted to gray scale images. Fig. 12 is a fraction of the positive samples for *bottle*.

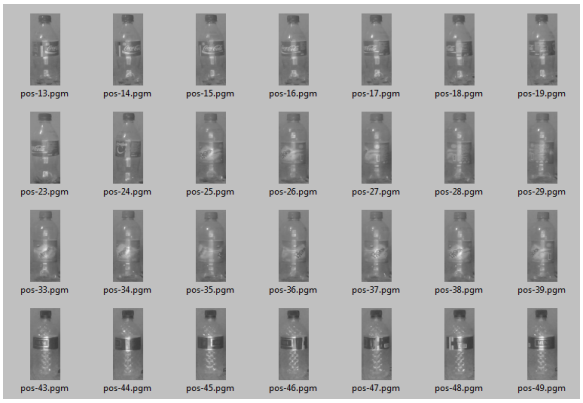


Fig. 12. A fraction of positive samples of *bottles*

Fig. 13 shows some negative samples we collected. It is a must that these images do not contain any of the positive images. These may be any other images which contain background objects.

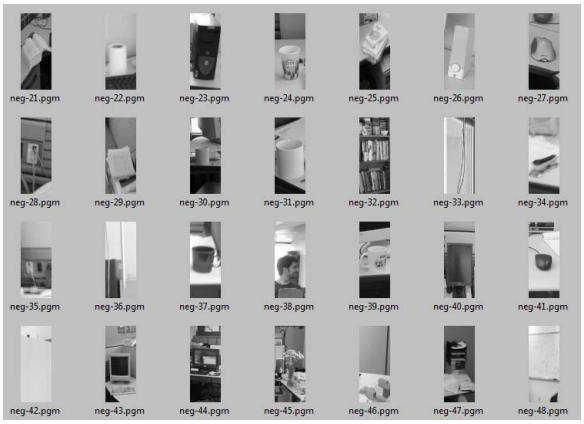


Fig. 13. A fraction of negative samples for *bottles*

B. Sample creation

Here, we need to create description file for both positive and negative images. Positive description file contains bounding rectangles of object: x, y, width, height and full path to positive images and negative description file only contains full path to negative images. After making description file, it needs to create a vector file based on positive description file using *opencv_createsamples*. This whole process is known as sample creation.

C. Training the classifier

The next step is the training of Haar-like cascade classifier. Using *opencv_traincascade*, we trained the cascade classifier. We set the minimum desired hit rate at 99.5%, and the maximum desired false alarm rate at 50% for each stage of the cascade of classifiers. An average of 20-stage cascade is trained for each object class. This training generates a XML file which is used in our detection framework to detect particular object class. Here, Gentle AdaBoost is used for learning classifiers. It combines weak classifiers into strong classifier. A classifier using an additive model is defined as in (2).

$$F(x) = a_1 f_1(x) + a_2 f_2(x) + a_3 f_3(x) + ... \quad (2)$$

Where, F stands for the strong classifier, x is the feature vector, a is the weight and f is the weak classifier respectively.

VII. Performance Evaluation

To evaluate the overall system, we have performed experiments for various cases. We have also calculated the accuracy of the system in real environment. Accuracy may be defined by the ratio between the total number of sample input object images and total number of detected objects. The accuracy usually represented as the percentages. We describes there cases for three different user requirements. One for object detected by color, one for object detection by size and another for object detection by position respectively.

A. Case 1

In the first example shown in Fig. 14, the user wanted the green glass in the scene. In this case, there were two different color glasses in the scene and system asked the user about color. Here, conversation between the system and user is as follows:

System: I found 5 objects: 2 bottles, 2 glasses, 1 bowl, 2 balls and 1 mug.
Which one do you want?
User: Glass.
System: There are 2 of it. Which one?
User: Green.
System: I got it.

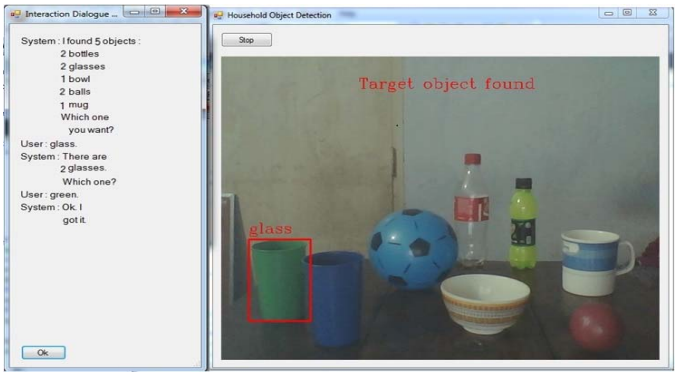


Fig. 14. Implementation of target object detection by color

B. Case 2

In the second example, shown in Fig. 15, the user wanted the small ball. In this case, there were two balls of different size and according to user feedback system detected the small ball. Here, conversation between the system and user is as follows:

System: I found 5 objects: 2 bottles, 2 glasses, 1 bowl, 2 balls and 1 mug.
Which one do you want?
User: Ball.
System: There are 2 of it. Which one?
User: Small.
System: I got it.

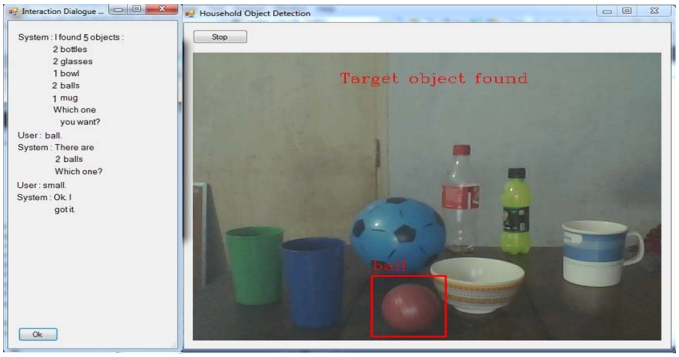


Fig. 15. Implementation of target object detection by size

C. Case 3

In the third example shown in Fig. 16, the user wanted the bottle second from left. In this case, system detected the particular bottle which is in the second position from left. These experimental results have confirmed that the system can work as expected. Here, conversation between the system and user is as follows:

System: I found 4 objects: 5 bottles, 2 glasses, 1 bowl, and 1 mug.
Which one do you want?
User: Bottle.
System: There are 5of it. Which one?
User: 2nd from left.
System: I got it.

Fig. 16. Implementation of target object detection by position

Graphical representation of the accuracy of proposed system based on the above three cases is shown in Fig. 17. It shows that the detection accuracy is greater when describing object by position and size than describing by color.

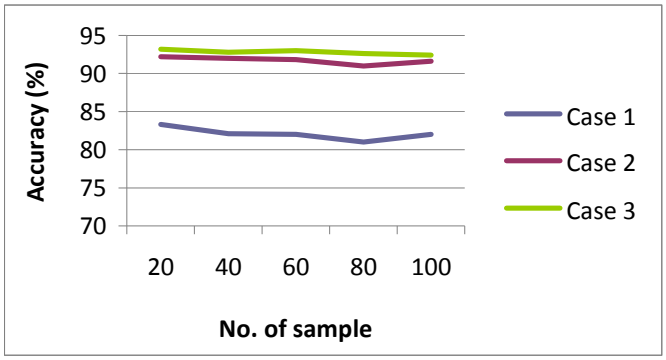


Fig. 17. Graphical representation of the accuracy of proposed system

After implementing the proposed system, we apply it on different formation of household objects and analyze the result considering detection accuracy and false positive. Detection accuracy indicates how accurate the detection is. Table II shows the performance evaluation of the system.

TABLE II: PERFORMANCE EVALUATION OF THE SYSTEM

Object Type	No of Sample	Sample Detected	Samples Missed	False Positive	Accuracy
Bottle	90	81	9	10%	90%
Glass	75	67	8	10.7%	89.3%
Mug	55	49	6	10.9%	89.1%
Bowl	50	44	6	12%	88%
Ball	60	53	7	11.7%	88.3%

False positive is a result that indicates a given condition has been fulfilled, when it actually has not been fulfilled. In our work, when system shows an object but actually there is no object that is a false positive. Performance evaluation indicates that the recognition rate varies a little bit depending on different object class and user preferences (expressions to describe an object).

VIII. CONCLUSION

Our primary aim was to detect specific household object from multiple objects in real time. Object detection in real time is relatively harder because it requires the skill to differentiate one object from the other, though they may be of same type and should work fast. Such problem is very difficult for machines, if they do not know about the various possibilities of objects. The results show that humans typically describe objects using one of multiple colors, size or position. Implementing this, we have made a system that can detect a specific household object using these expressions. The overall experimental result including subjective and objective experiment shows that the project is functioning quite well. Finally we can say that the system can detect specific household object from multiple objects in real time more accurately using human interaction. The future recommendations are to add hardware support which can move camera in case it needed in complex environment, to add more expressions to describe a household object.

References

[1] P. Dollar, Z. Tu, P. Perona, and S. Belongie, “Integral channel features,” In Proc. British Machine Vision Conference, 2009.

[2] S. Paisitkriangkrai, C. Shen, and J. Zhang, “Incremental training of a detector using online sparse eigendecomposition,” IEEE Trans. on Image Processing, vol. 20, no. 1, pp.213–226, 2011.

[3] C. Shen, S. Paisitkriangkrai, and J. Zhang, “Efficiently learning a detection cascade with sparse eigenvectors,” IEEE Trans. on Image Processing, vol. 20, pp. 22–35, 2011.

[4] J. Gall, A. Yao, N. Razavi, L. V. Gool, and V. Lempitsky, “Hough forests for object detection, tracking, and action recognition,” IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 33, no. 11, pp. 2188–2202, 2011.

[5] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” In Proc. the Conference on Computer Vision and Pattern Recognition, vol. 1, p. 511, 2001.

[6] M. Takizawa, Y. Makihara, N. Shimada, J. Miura, and Y. Shirai, “A Service Robot with Interactive Vision- Objects Recognition using Dialog with User,” In Proc. First International Workshop on Language Understanding and Agents for Real World Interaction, Hokkaido, 2003.

[7] T. Kawaji, K. Okada, M. Inaba, H. Inoue, “Human Robot Interaction through Integrating Visual Auditory Information with Relaxation Method,” In Proc. International Conference on Multisensor Fusion on Integration for Inteligent Systems, Tokyo, pp 323–328, 2003.

[8] K. Komatani, T. Kawahara, R. Ito and H. G. Okuno, “Efficient Dialogue Stategy to Find User’s Intended Items from Information Query Results," In Proc. 19th International Conference on Computational Linguistics, Taipei, pp. 481-487, 2002.

[9] R. Kurnia, M. A. Hossain, A. Nakamura, Y. Kuno, “Generation of efficient and user-friendly queries for helper robots to detect target objects,” Advanced Robotics 20(5): 499-517, 2006.

[10] Y. Yamakata, T. Kawahara and Hiroshi G. Okuno, “Belief Network Based Disambiguation of Object Reference in Spoken Dialogue System for Robot,” In Proc. ISCA workshop on Multi-modal Dialogue in Mobile Environment, Alaska, 2002

[11] A. Mansur and Y. Kuno, “Specic and class object recognition for service robots through autonomous and interactive methods," in IEICE - Trans. Inf.Syst., vol. E91-D, no. 6, pp. 1793-1803, 2008.

[12] A. Mansur, K. Sakata, T. Rukhsana, Y. Kobayashi, Y. Kuno,” Human Robot Interaction Through Simple Expressions for Object Recognition,” In: Proc. 17th IEEE RO-MAN, pp. 647–652, 2008.