Generation of Efficient and User-friendly Queries for Helper Robots to Detect Target Objects

RAHMADI KURNIA, MD. ALTAB HOSSAIN, AKIO NAKAMURA, and YOSHINORI KUNO

Department of Information and Computer Sciences, Saitama University,

255 Shimo-Okubo, Sakura-ku, Saitama-shi 338-8570, Japan

Email :{kurnia, hossain, nakamura, kuno}@cv.ics.saitama-u.ac.jp

Abstract—We are developing a helper robot that carries out tasks ordered by users through speech. The robot needs a vision system to recognize the objects appearing in the orders. However, conventional vision systems cannot recognize objects in complex scenes. They may find many objects and cannot determine which is the target. This paper proposes a method of using a conversation with the user to solve this problem. The robot asks a question to which the user can easily answer and whose answer can efficiently reduce the number of candidate objects. It considers the characteristics of features used for object identification such as the easiness for humans to specify them by word, generating a user-friendly and efficient sequence of questions. Experimental results show that the robot can detect target objects by asking the questions generated by the method.

Keywords: robot vision; dialog generation; image segmentation; human-robot interface

1. INTRODUCTION

Helper robots or service robots in welfare domain have attracted much attention of researchers for the coming aged society [1][2]. Multimodal interfaces [3][4][5] are considered good interface means for such robots. Thus, we are developing a helper robot that carries out tasks ordered by the user through voice

and/or gestures [6][7][8]. In addition to gesture recognition, such robots need to have vision systems that can recognize the objects mentioned in speech.

It is, however, difficult to realize vision systems that can work in various conditions. Thus, we have proposed to use the human user's assistance through speech [6][7][8]. When the vision system cannot achieve a task, the robot makes a speech to the user so that the natural response by the user can give helpful information for its vision system. In our previous work, however, we inexplicitly assume that the scene is relatively simple so that the vision system detects one or at most a few regions (objects) in the image. Thus, even though detecting the target object may be difficult and need the user's assistance, once the robot has detected an object, it can assume the object as the target. However, in actual complex scenes, the vision system may detect various objects. The robot must choose the target object among them, which is a hard problem especially if it does not have much a priori knowledge about the object. This paper tackles this problem. The robot determines the target through a conversation with the user. The point of research is how to generate a sequence of utterances that can lead to determine the object efficiently and user-friendly. This paper presents such a dialog generation method. It determines what and how to ask the user by considering the image processing results and the characteristics of object (image) attributes.

There has been a great deal of research on robot systems understanding the scene or their tasks through interaction with the user [9][10][11][12][13][14][15], dating back to the work by Winograd [16]. Especially, the settings in [13] [14] are similar to ours. The robot makes quires to the user to understand the target objects that the user has in mind. Yamakata et al. [13] have presented a probabilistic reasoning method based on a belief network of the object reference. Inamura et al. [14] have proposed a method based on a Bayesian Network, which examines the certainty factors to determine what features should be used to narrow down the target. Our research is similar from outlook to these studies. However, ours is different in that its main concern is the problems and issues of computer vision. Conventional systems mainly consider dialog generation at the language level, treating image features and attributes equally. Thus, as long as the certainty factors or any other values alike for features are the same, the systems acts in the same way regardless what kind of features are involved. However, all image features and attributes are not the same in their characteristics. For example, humans can easily describe some features by word while

not others. Some features should be treated differently depending on the existence of other features. As mentioned before, our purpose is to develop a vision system that can work in complex real world with the help of human interaction. Thus, we investigate the characteristics of image features from various viewpoints and propose a method of generating efficient and user-friendly dialogs based on the investigation results. As to the related work on computer vision, we need to mention the work by Roy and Pentland [17]. They have proposed a system that can learn words by interacting with a person. The work is similar in that a machine system recognizes unknown objects through speech interaction. However, their work assumes, as in our previous work, that the target object is presented before the system, thus it does not need to determine which is the target.

This paper presents the dialog generation method and proves its usefulness thorough experiments. Section 2 describes the basic ideas about the method. Section 3 shows the characteristics of image features from the viewpoint of dialog generation. Section 4 presents a dialog generation method. Section 5 described the image processing modules used in the system. The section may deviate from the main track of dialog generation. However, It is an important part to realize the system. Section 6 shows experiments. Section 7 concludes the paper.

2. BASIC IDEAS

This section briefly describes the basic ideas behind our dialog generation method.

We represent objects by their attributes such as color and shape. The vision system tries to detect regions with the attributes of the target object. For example, assuming that 'apple' is represented as a red round object. If the user asks the robot to get the apple, the robot initiates color segmentation and shape detection processes. If it can find a red round object, it asks the user for confirmation through speech. Otherwise, it explains the current vision results through speech, expecting that the user's reply may help to recognize the object.

In [6], we consider the cases where the robot has a priori knowledge about target objects and the failure of vision comes from the difference between the current object attributes and the stored knowledge. For example, in the apple's case mentioned above, the robot cannot detect an apple if the apple in the scene

is a green apple. In this case, the robot tells the user that it cannot find a red object but detect a green round object. From this, the user knows that the robot does not know the existence of green apples. We can expect him/her to say something about green color to the robot. In [8], we propose an object recognition method that learns appropriate vision processes depending on the environment through its use with interaction with the user. We also assume that the robot knows a priori knowledge about target objects.

In this paper, we deal with objects with no a priori knowledge. The user may say object names that the robot does not know what they are, or he/she just mentions them using deictic words such as 'that' [18]. We would like to enable the robot to work in such situations. In addition, more importantly, we consider actual complex situations where it is difficult to choose a target object among many objects. In our previous work, we inexplicitly assume that the scene is simple so that the vision system detects one or at most a few regions (objects) in the image. Thus, even though detecting the target object may be difficult, once something has been detected, the system can assume it as the target. However, in actual complex scenes, the vision system may detect various objects, especially if it does not have a priori knowledge about the object.

As mentioned earlier, we represent an object as a set of attributes and recognize it by finding a region with the attributes. Thus, if the user gives the robot the information about some attributes of the target object, it can remove the objects that do not satisfy the attributes, reducing the number of candidates for the target object. In other words, the robot can identify the target object by asking the user for the attributes of the target object. However, if it asks him/her all the information at once, he/she may find it difficult to answer. It is easy for humans to answer to short simple questions. On the other hand, it is not good for users if the robot needs too many questions, even if each is simple, to identify the target object. The point is, therefore, how to generate a sequence of utterances leading to identify the target objects efficiently and user-friendly.

In the current implementation, we impose the following limitations on the robot's utterances. The robot asks the user a question about one attribute at a time. The type of question is either 'what-question' or 'yes/no-question' except in some special cases. We adopt this one-at-a-time approach not only because this is simple but also because this is actually a useful strategy. Suppose we have ten candidates, among which

only one is red and big. Consider the case that the robot asks, "Is the target object red and big?" If the answer is "Yes," this question is really efficient and user-friendly. However, if the answer is "No," this question can only reduce the number of candidates just by one. Since we assume that we do not have any a priori knowledge about the object, the probability of the yes-case is 10% and that of the no-case is 90%. If we use multiple attributes at a time, we can reduce the candidates if the answer is "Yes." However, the probability of getting the yes answer reduces. The expected reduction number of candidate objects is the same. If the effect is the same as mentioned above, it is better to keep things simple. In addition, if the robot asks about multiple attributes at the same time, this may put some cognitive burden on the person when the answer is "No." He/she may wonder how to answer to the question depending on the situation, such as that all the attributes are negative or that a part of the attributes is negative. He/she may respond by complex statements. This may make speech recognition difficult for the robot. For example, he/she may say, "That is red but I can't say it's big," when the target object is red but not big in the above example case. The one-at-a-time approach is good for both humans and the robot.

What question that the robot should ask depends on the current vision results and the characteristics of attributes. If all the detected regions are different in a particular attribute, asking the attribute may help much to determine the target. For example, if all the regions in the initial segmentation result are different in color, it may be appropriate to ask, "What color is it?" However, even if all the objects are different in shape, if they are of irregular shape, it is not good to ask, "What shape is it?" The user finds it difficult to answer to such a question by speech. We need to consider such characteristics of features in generating utterances. We define the characteristics of each feature from four viewpoints: vocabulary, distribution, uniqueness, and relativity. The next section describes details of these characteristics. Regions in the image segmentation result are classified into groups for each feature. For example, they are classified into typical color groups in the case of color feature. Then, the system chooses the feature about which it asks the user by considering the distribution of the regions in terms of each feature and its characteristics.

3. FEATURE CHARACTERISTICS

We consider the characteristics of features to determine which feature the robot uses and how to use it from the following four viewpoints. We make a binary decision from each viewpoint for each feature. In the current implementation, we use four features: color, size, position, and shape. Thus, they are used in the following explanation as examples. We can consider the characteristics of other features in the same way. Note that we consider features in 2-D images in the current implementation. Size, position, and shape are not those in the 3-D world but those in 2-D images.

1. Vocabulary

Humans can easily describe some features by word but cannot do so for other features. If we can represent a particular feature easily by word for any given object, we call it a *vocabulary-rich* feature. The robot can ask relatively complex questions such as 'what-type' questions for a *vocabulary-rich feature* since we can easily find an appropriate word for answer. For example, we have rich vocabulary for color description: such as, red, green, blue, etc. When the robot asks what color it is, we can easily give an answer. Position is also a *vocabulary-rich* feature. We have a large vocabulary to express position such as left, right, upper, and lower. Size is not this type of feature. We do not have much vocabulary to describe size. Although we have a rather large vocabulary about shape, it is not a *vocabulary-rich* feature by our definition, since we cannot express irregular shapes easily by word.

2. Distribution

Although we consider features of each object independently, we may find it difficult to express some features by word depending on the spatial distribution of objects. We call a feature with this problem a *distribution-dependent* feature. Position is a *distribution-dependent* feature. If several objects exist close together, it is difficult to specify the position of each object. Color, size, and shape are not such features.

3. Uniqueness

If the value of a particular feature is different for each object, we call it a *unique* feature. Position can be a *unique* feature since no multiple objects share the same position.

4. Absoluteness/Relativeness

If we can describe a particular feature by word even if only an object exists, we call it an absolute feature. Otherwise, we call it a relative feature. Color and shape are absolute features in general. (Although we can compare objects in terms of such features, we do not consider such cases here.) Size and position are not absolute features but relative features. We say 'big' or 'small' by comparison with other objects. Positional relation between multiple objects is also relative. Although we may seem to be able to specify the object's position in an image absolutely, this is because we consider the position relative to the image frame.

Table 1 summarizes the characteristics of the features used in the current implementation.

Table 1. Features and their characteristics

Characteristic	Color	Size	Position	Shape
Vocabulary	√	-	√	-
Distribution	-	-	√	-
Uniqueness	-	-	√	-
Absoluteness	√	Relative	Relative	√

4. DIALOG GENERATION

The basic strategy for generating a dialog is 'ask-and-remove'. The robot asks the user about a certain feature. Then, it removes unrelated objects from the detected objects using the information given by the user. It iterates this process until only an object remains.

The robot applies color segmentation, then obtaining features for each segmented foreground region. Section 5 describes these processes. The robot divides the current situation into two cases according to the number of detected regions: the few-object case where the number of regions is equal to or less than three, and the many- object case where that is greater than three.

4.1. Many-Object Case

When the number of objects is large, it may be difficult to use *distribution-dependent* features since multiple objects may closely exist. It may be also better to avoid using *relative* features since it may be difficult to choose some objects by comparison with many other objects using such features. Thus, we mainly consider *vocabulary-rich* features and *absolute* features when the number of objects is greater than three. We consider *unique* features only when the other features cannot work, because in the current implementation, position is the only *unique* feature, and it is a *distribution-dependent* feature.

The robot generates its utterances for dialog with the user as follows. First, it classifies the features of all regions into classes. For example, it assigns a color label to each region based on the average hue value of the region. How to classify the data is determined for each feature in advance. For color, it classifies them into seven colors: blue, yellow, green, red, magenta, white, and black.

Then, the robot computes the percentage of the number of objects in each class to the total number of objects. It classifies the situation of each feature into three categories depending on the maximum percentage: the variation category, the medium category, and the concentration category. The variation category is the case where the maximum percentage is less than 33% (1/3). The concentration category is the case where that is more than 67% (2/3). The medium category is the case that does not belong to both categories, that is, the maximum percentage is from 33% through 67%. These percentage values are experimentally determined.

If the robot can obtain information about any feature that falls under the variation category, the information can reduce many unrelated objects among the regions (object candidates). Therefore, the first rule for determining what feature the robot chooses for its question to the user is to give a priority to the variation category features. If no such feature exists, the medium category features are given the second priority and the concentration category features the last.

1. Case with variation category feature

If there are any variation category features, the robot asks the features to the user. If the present features classified into the variation category include a *vocabulary-rich* feature, the robot asks the user 'what-type' question about the feature. For example, if the color feature satisfies the variation category

condition, the robot asks, "What is the color of the target object?" since color is a *vocabulary-rich* feature. If there are multiple *vocabulary-rich* features, the first priority is given to the feature with the smallest maximum percentage.

If there is no *vocabulary-rich* feature, the robot needs to adopt *absolut*e features. Since they are not *vocabulary-rich* features, the user may find it difficult to answer the question if the robot asks a 'what-type' question. Thus, the robot examines whether or not each region can be described easily by word in terms of the feature. If all regions satisfy this, the robot adopts a 'what-type' question. Otherwise, it uses a multiple choice question such as, "Is the target object A, B, or others?" where 'A' and 'B' are features that can be expressed by word easily. For example, in the case of shape, the robot may asks, "Is the target object a circle, a rectangle, or others?" There could be a case where all regions are hard to be expressed by word. However, this does not happen in the current system. It classifies the regions into classes that can be expressed by word; and it assigns the label 'others' to the regions that cannot be expressed by word. Thus, the number of regions with the 'others' label should be less than one third of the total number if the feature is classified into the variation category.

2. Case with medium category features

If no features fall under the variation category but any under the medium category, the robot considers to use the features in the medium category. In this case, the robot uses a 'yes/no-type' question. It is easier for humans to answer to 'yes/no-type' questions than to 'what-type' questions; and even if the robot uses 'what-type' questions, it cannot reduce the number of regions much in this case. The robot generates a question such as, "Is the target object A?" where 'A' is the label of the feature with the largest percentage. An example is, "Is the target object red?" The robot can reduce the number of candidates into half on average by one question. If there are multiple such features, the robot gives them priorities according to the order fixed in advance. We determine the priority in the order that we can obtain reliable information. In the current implementation, color comes first followed by shape.

3. Case with concentration category features

This is the case where all features are classified into the concentration category, which means that all regions (objects) are similar in several respects. Thus, the robot plans to use *unique* features. The robot

asks a 'yes/no-type' question about *unique* features. In the current implementation, position is the only *unique* feature. An example question is, "Is the target object on the right?" The robot computes the spatial distribution pattern of the objects. It computes the distances among the objects, then classifying the objects into groups with those closely located. The spatial distribution patterns are defined by the positional relationships among these groups. For each pattern, a word or phrase representing the pattern, 'on the right' in the above example, is determined in advance. The robot chooses the word or phrase for the current pattern to generate an utterance.

When we use position, we need to consider two things. One is that position is a *distribution-dependent* feature. The other is that in the current implementation we consider only 2-D position, that is, the position in the image. However, the user does not know the position in the image but knows that in the 3-D world. We assume that the user uses words specifying positional relationships, such as 'right' and 'left' by considering the robot's camera direction. Thus, 'right' means the right part in the image, and 'close' means the lower part in the image. However, such interpretation may be wrong and asking the user to translate 3-D positional relation into that of 2-D does not conform to the purpose of this research. To solve these problems, we are planning to specify positional relationships with respect to some distinguished objects in the scene. For example, if the robot finds a red object in the scene where no other red objects can be seen, it asks the user, such as, if the target object is on the right of the red object. This is left for future work.

4.2. Few-Object Case

If the number of objects is at most three, we can use *relative* features and *distribution-dependent* features in addition to the features used in the many object case. In the current implementation, only position is a *distribution-dependent* feature and it also a *relative* feature. Thus, we consider only *relative* features.

In this case, the robot first tries the first part of the many object case with variation category features. That is, if any *vocabulary-rich* feature falls in the variation category, the robot asks a 'what-type' question about the feature.

Then, if any vocabulary-rich or *absolute* feature falls in the medium category, the robot adopts a question shown in the case with medium category features for many objects.

If all of the above does not hold to the current situation, the robot examines the degree of variation for each relative feature, then giving priorities in the order of the larger variation. The degree of variation is defined by the normalized difference between the maximum and the minimum. We give some weights experimentally determined to the degrees of variation to compare those for different features. The robot asks a 'yes/no-type' question about the highest priority feature, such as "Is the target object big?"

Although we use only four features in the current implementation, we can easily add a new feature in terms of dialog generation. (Developing a feature extraction method is another issue.) If we would like to do so, we analyze the feature from the viewpoints described in Section 3. Then, we can determine how to use the new feature in the dialog generation based on the analysis result. This is one of the advantages of systematic treatment of image features.

5. IMAGE PROCESSING

This section describes our image processing modules used in the experiments. In the current implementation, we first apply color segmentation and compute four features for each foreground region in the segmentation result: color, shape, position, and size. The position of an object is the centroid of the region. The size is the number of pixels of the region.

5.1. Color Segmentation

Color segmentation plays the most important role in realizing our system. We need to extract object regions robustly from images taken under various conditions so that the colors of the regions can match those perceived by humans. We propose an image segmentation method satisfying this need. The method uses a robust approach of features space method: the mean shift algorithm [19, 20] combined with HSI (Hue, Saturation, and Intensity) color space for color image segmentation. The mean shift algorithm can analyze a complex multimodal feature space and delineate arbitrarily shaped clusters. Although the mean shift algorithm and HSI color space can be separately used for color image segmentation, they surely fail to segment images when illumination condition will change. To solve this problem, we use the mean shift

algorithm as an image preprocessing tool to reduce regions and the numbers of colors used and then use the HSI color space for merging regions originating from single objects.

Our method consists of the following parts:

• Apply the mean shift algorithm into a real image to reduce colors and divide it into several regions.

• Merge the regions based on H, S, I components of HSI color space.

• Filter the result using the median filter.

• Eliminate the small regions using the region growing algorithm.

The input image is first analyzed using the mean shift algorithm. The mean shift algorithm can be described as follows:

Let {Xi}i=1,…,n be a set of n data points in a d-dimensional Euclidian space R^d, the multivariate kernel density estimator with kernel K and window radius (band-width) h is defined as follows ([21]):

$$\hat{f}(x) = \frac{1}{nh^d} \sum_{i=1}^{n} k(\frac{x - X_i}{h}) \qquad (1)$$

The kernel function should satisfy some conditions [22]. The Epanechinkov kernel [21] is one of the optimum kernels, which yields the minimum mean integrated square error (MISE):

$$k_e(x) = \begin{cases} \frac{1}{2} c_d^{-1} (d+2)(1 - x^T x) & \text{If } x^T x < 1 \\ 0 & \text{Otherwise} \end{cases} \qquad (2)$$

Where c_d is the volume of the unit d-dimensional sphere, e.g., $c_1=2$, $c_2=\pi$, $c_3=4\pi/3$.

Thus, the density gradient estimate of the Epanechinkov kernel can be written as:

$$\hat{\nabla} f(x) \equiv \nabla \hat{f}(x) = \frac{1}{nh_d} \sum_{i=1}^{n} \nabla k(\frac{x - X_i}{h}) \qquad (3)$$

Equation (3) can be rewritten as:

$$\hat{\nabla} f(x) = \frac{n_x}{n(h^d c_d)} \frac{d+2}{h^2} \left(\frac{1}{n_x} \sum_{X_i \in S_h(x)} (X_i - x) \right) \quad (4)$$

Where the region $S_h(x)$ is a hypersphere of radius h, having the volume $h^d c_d$, centered at x, and containing n_x data points.

The mean shift vector $M_h(x)$ is defined as:

$$M_h(x) \equiv \frac{1}{n_x} \sum_{X_i \in S_h(x)} (X_i - x) = \frac{1}{n_x} \sum_{X_i \in S_h(x)} X_i - x \quad (5)$$

From equations (4) and (5), we get:

$$M_x(x) \equiv \frac{h^2}{d+2} \frac{\hat{\nabla} f(x)}{\hat{f}(x)} \quad (6)$$

The mean shift is an unsupervised nonparametric estimator of density gradient and the mean shift vector is the difference between the local and the center of the window. Using the mean shift procedure to analyze clusters has the following two main advantages:

• The method is application independent and does not need any a priori assumptions such as the number of clusters or the shape of the clustering regions. Hence it can be easily applied to solve clustering problems across multiple domains.

• It is efficient and adaptive because, as pointed out by Comaniciu and Meer [19], the mean shift vector always points to the direction of maximum increase in the density and the shift steps are large in low-density regions and small near local maximums. Furthermore, the computation cost is not high compared to other clustering methods. This is a desirable feature to real-time or interactive applications.

The input image may contain many colors and several regions. Applying the mean shift algorithm, we can significantly and accurately reduce the number of colors and regions. Thus, the output of the mean shift algorithm is several regions with a fewer numbers of colors in comparison with the input image.

These regions, however, do not imply that each comes from a single object. The mean shift algorithm may divide even a single color object into several regions with more than one color. To remove this ambiguity, we use the Hue, Saturation and Intensity components of the HSI color space to merge the homogeneous regions which likely come from a single object, as Hue is invariant to certain types of highlights, shading, and shadows. For transformation of the image from RGB to HSI space, we use the following equations:

$$H = \arccos\left(\frac{0.5((R-G)+(R-B))}{\sqrt{(R-G)^2+(R-B)(G-B)}}\right) \tag{7}$$

If B>G then: H: $=360^0$-H

$$S = 1-3\frac{\min\{R,G,B\}}{R+G+B} \tag{8}$$

$$I = \frac{(R+G+B)}{3} \tag{9}$$

For homogeneous regions, we use threshold values for each component of HSI. We use the histograms of each component to select the appropriate threshold. The threshold values are selected dynamically based on the illumination condition of the image.

Then we use the median filter as image post-processing. This may help to smooth the image boundary and also helps to reduce the unwanted regions. Finally, we also use the region growing procedure as another image post-processing procedure to avoid over segmentation or remove small highlights from objects. This procedure removes all regions smaller than a given area analyzing their 4-connected neighborhood.

In the current implementation, we restrict to use seven colors: green, blue, yellow, red, magenta, black and white so that humans can perceive and describe colors without any ambiguity. The system gives these color labels to detected regions using the color ranges prepared in advance. Regions whose color is out of

any range are labeled as 'other colors'. Color is not asked for these regions. Since humans have rich vocabulary of color, this restriction should be eased. We are now planning to make the system acquire additional vocabulary of color through interaction with humans. This is left for future work.

5.2. Shape Detection

We compute the shape factor S for each segmented region by,

$$S = 4\pi \frac{area}{(perimeter)^2}$$

We classify the regions into shape categories by this value. If it is around 1, the shape is a circle; around 0.8, a square, 0.6, a triangle, less than 0.6, an irregular shape.

6. EXPERIMENTAL RESULTS

We performed experiments for various cases. Here, we show three typical example cases.

In the first example shown in Fig. 1, the user wanted the green ball in the scene. The dialog in this case was as follows.

Robot: *What is the color of the target object?*

User: *Green.* (Two objects remain as in Fig. 1 (b).)

Robot: *Is the target object in the left side?*

User : *No.*

The robot understood the object shown in Fig. 1 (c) was the target object.

In this case, there were various color objects in the scene and color is a vocabulary-rich feature. Thus, the system asked the user about color using a what-type question.

In the second example shown in Fig. 2, the user wanted the yellow ball. The dialog in this case was as follows.

Robot: *Is the target object a circle, rectangle, triangle or others?*

User : *Circle.* (Two circle objects remain as in Fig. 2(b).)

Robot: *Is the target object in the left side?*

User : *No*

The robot understood the object shown in Fig. 2 (c) was the target object.

In this case, there were objects of various shapes. Since shape is not a vocabulary-rich feature, the system adopted a multi-choice type question.

In the third example shown in Fig. 3, the user wanted the blue book. The dialog in this case was as follows.

Robot: *Is the target object yellow?*

User : *No.* (The system removes the yellow objects as in Fig. 3(b).)

Robot: *Is the target object blue?*

User : *Yes.* (Three objects remain as in Fig.3(c).)

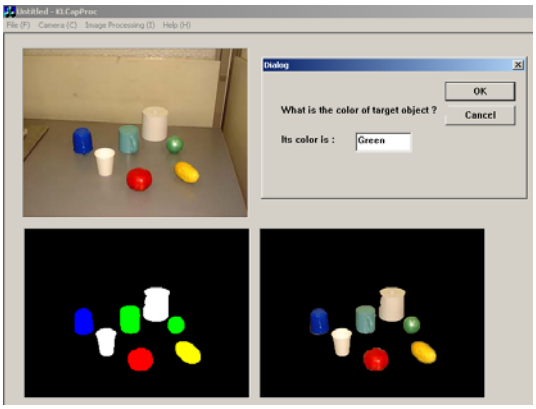
Robot: *Is the target object big?*

User : *Yes.*

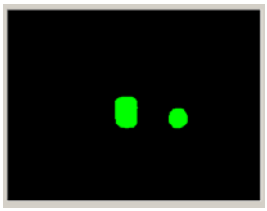
The robot understood the object shown in Fig. 3 (d) was the target object.

In this case, there were many yellow objects. Thus the system asks the user about yellow color objects using a yes-no type question.

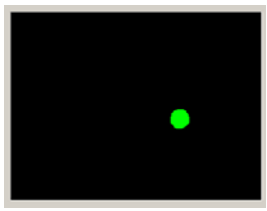
These experimental results have confirmed that the system can work as expected. It can generate appropriate questions depending on the situation.



(a) Input image (top left); Color segmentation (bottom left); Foreground objects (bottom right).

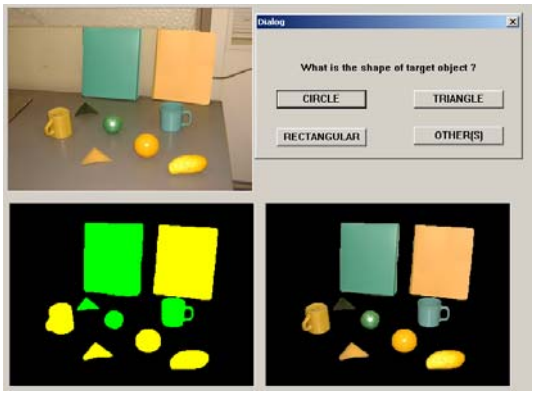


(b) After the 1st answer.



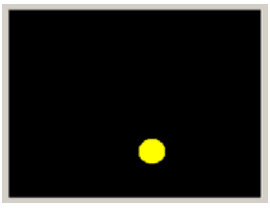
(c) Final result (target object).

Fig. 1. Experimental result 1.



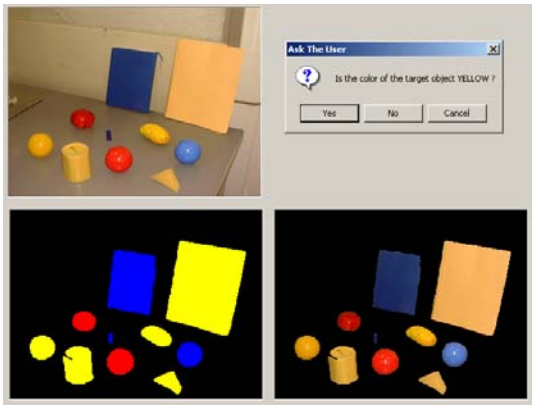
(a) Input image (top left); Color segmentation (bottom left); Foreground objects (bottom right).

(b) After the 1st answer.

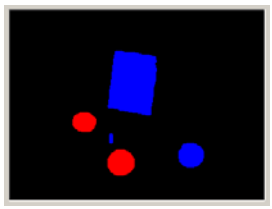


(c) Final result (target object).

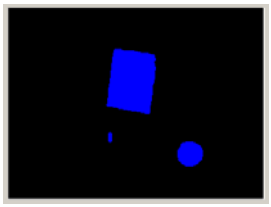
Fig. 2. Experimental result 2



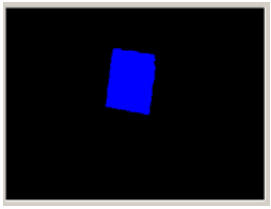
(b) Input image (top left); Color segmentation (bottom left); Foreground objects (bottom right).



(b) After the 1st answer.



(c) After the 2nd answer.



(d) Final result (target object).

Fig. 3. Experimental result 3.

Then, we performed an experiment using simulation data to examine the efficiency of the method. We classified color feature into seven classes (green, blue, yellow, red, white, magenta and black), shape into four (circle, square, triangle and irregular shape), size into two (big, small), and position into four (left, right, top, and bottom). We generated all possible combinations of feature variations for a given number of objects. We recorded the number of questions that the system needed to locate the target object for every generated case.

Fig. 4 shows the experimental result. The horizontal axis represents the number of objects in an image. The vertical axis represents the average number of questions. The smooth curve in the figure indicates $\log_2 n$ where n is the number of objects. If the method uses only yes/no type questions, the average number of questions may be around this curve. The average numbers obtained by the method are below the curve. This shows the effectiveness of the method, although it can be expected since the method uses what-type

questions. However, note that the method uses what-type questions only when humans can answer them easily. The method satisfies both effectiveness and user-friendliness.

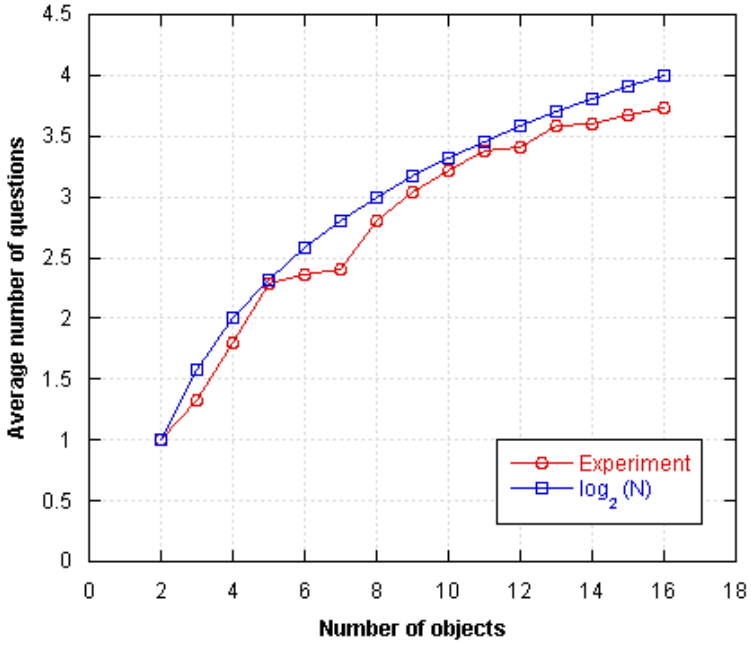


Fig. 4. Simulation experiment.

Finally, we compared the system with humans. We prepared six scenes, two of which are shown in Fig. 5. A person chose an object as the target in each scene. We asked six other participants to guess what was the target object by asking 'yes/no-type' and 'what-type' questions to the person. Our system also generated a sequence of questions to ask the person.

Figure 6 shows the result. It shows the average, the maximum, and the minimum number of questions necessary for the human participants, and the number of questions that the system asked. Our system always needed less number of questions compared with those by the humans. The result proves that the system can generate efficient questions.

(a) (b)

Fig. 5. Experimental scenes. (a) scene 1 (b) scene 4 in Fig 6.

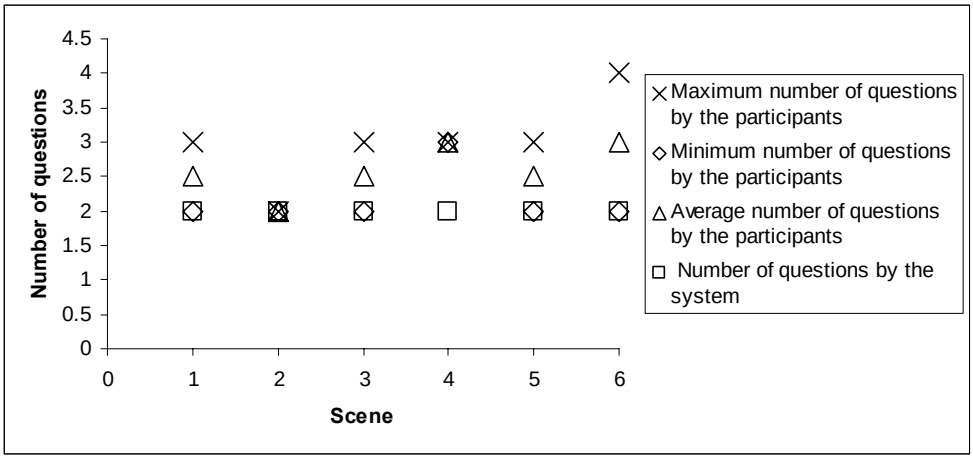


Fig. 6. The number of questions necessary for the human participants and the system.

7. CONCLUSION

We are developing a helper robot that carries out tasks ordered by users through speech. Such a robot needs a vision system to recognize the objects appearing in the orders. However, conventional vision systems cannot recognize objects in complex scenes. They may find many objects and cannot determine which is the target. We have proposed to use a conversation with the user to solve this problem. The point of research is how to generate a sequence of questions that can lead to determine the object efficiently and

user-friendly. The vision process starts with image segmentation. The robot tries to identify objects using the image features of each segmented region. We analyze the characteristics of image features from four viewpoints: vocabulary, distribution, uniqueness, and relativity. We show that the robot can generate efficient user-friendly questions by considering these characteristics. Experimental results have proved the effectiveness of the method.

In this paper, we assumed that each segmented region in images corresponds to a different object. However, we cannot always expect such perfect segmentation results. An object may be divided into multiple regions, or multiple objects may be merged into a region. We are now working on this problem

ACKNOWLEDGEMENT

This work was supported in part by the Ministry of Education, Culture, Sports, Science and Technology under the Grant-in-Aid for Scientific Research (KAKENHI 14350127).

REFERENCES

[1] M. Ehrenmann, R. Zollner, O. Rogalla, and R. Dillmann, Programming Service Tasks in Household Environments by Human Demonstration, in *Proc. International Workshop on Robots and Human Interactive Communication,* Berlin, pp.460-467 (2002)

[2] M. Hans, B. Graf, R.D. Schraft, Robotics Home Assistant Care-O-bot: Past-present-future, in *Proc. International Workshop on Robots and Human Interactive Communication* , Berlin, pp.380-385 (2002)

[3] G. A. Berry, V. Pavlovic, and T. S. Huang, Battle View: A Multimodal HCI Research Application, in *Proc. Workshop on Perceptual User Interfaces*, San Francisco, pp. 67-70 (1998)

[4] I. Poddar, Y. Sethi, E. Ozyildiz, and R. Sharma, Toward Natural Gesture/speech HCI: A Case Study of Weather Narration, in *Proc. Workshop on Perceptual User Interfaces*, San Francisco, pp. 1-6 (1998)

[5] R. Raisamo. A Multimodal User Interface for Public Information Kiosks, in *Proc. Workshop on Perceptual User Interfaces*, San Francisco, pp. 7-12 (1998)

[6] T. Takahashi, S. Nakanishi, Y. Kuno, and Y. Shirai, Human-robot Interface by Verbal and Nonverbal Communication, in *Proc. International Conference on Intelligent Robots and Systems*, Victoria, pp.924-929 (1998)

[7] M. Yoshizaki, Y. Kuno, and A.Nakamura, Mutual Assistance between Speech and Vision for Human-robot Interface, in *Proc. International Conference on Intelligent Robots and Systems* , Switzerland, pp.1308-1313 (2002)

[8] M. Yoshizaki, A. Nakamura, and Y. Kuno, "Vision-speech system adapting to the user and environment for service robots," in *CD-ROM of International Conference on Intelligent Robots and Systems*, Las Vegas, (2003)

[9] 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)

[10] 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)

[11] P. McGuire, J.Fritsch, J.J. Steil, F. Roothling, G.A. Fink, S. Wachsmuth, G. Sagerer, H. Ritter, Multi-modal Human Machine Communication for Instruction Robot Grasping Tasks, in *Proc. International Workshop on Robots and Human Interactive Communication* , Berlin, pp. 1082-1089 (2002)

[12] Kazunori Komatani, T. Kawahara, Ryousuke Ito and Hiroshi 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)

[13] Yoko 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)

[14] T. Inamura, M. Inaba, and H. Inoue, Dialogue Control for Task Achievement based on Evaluation of Situational Vagueness and Stochastic Representation of Experiences, in *Proc. International Conference on Intelligent Robots and Systems,* Sendai, pp. 2861-2866(2004)

[15] Anita Cremers, Object Reference in Task-Oriented Keyboard Dialogues, in *Multomodal Human-Computer Communication : System, techniques and experiments*. Springer, pp. 279-293, (1998)

[16] T. Winograd, *Understanding Natural Language*, New York: Academic Press (1972)

[17] D. Roy, B. Schiele, and A. Pentland, Learning Audio-visual Associations using Mutual Information, in *Proc. International Conference on Computer Vision, Workshop on Integrating Speech and Image Understanding*, Greece, (1999)

[18] Z. M. Hanafiah, C. Yamazaki, A. Nakamura and Y. Kuno, Human-robot Speech Interface Understanding Inexplicit Utterances using Vision, in *Proc. Conference on Human Factors in Computing Systems*, Vienna, pp.1321-1324 (2004)

[19] D. Comaniciu and P. Meer, Mean shift : A Robust Approach toward Feature Space Analysis, in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, pp. 603 – 619 (2002)

[20] Y. Cheng, Mean Shift, Mode Seeking, and Clustering, in *IEEE Transactions. Pattern Analysis and Machine Intelligence*, 17(8): p. 790-799 (1995)

[21] B.W. Silverman, *Density Estimation for Statistics and Data Analysis*, Chapman and Hall (1986)

[22] M.P. Wand and M. Jones, *Kernel Smoothing,* Chapman & Hall (1995)