Strategy for Displaying the Recognition Result in Interactive Vision

Yasushi Makihara, Jun Miura
Dept. of Computer-controlled Mechanical Sys.,
Graduate School of Engineering, Osaka Univ.,
2-1 Yamadaoka, Suita, Osaka, 565-0871, Japan
{makihara,jun}@cv.mech.eng.osaka-u.ac.jp

Yoshiaki Shirai, Nobutaka Shimada
Coll. of Information Science and Engineering,
Ritsumeikan Univ., 1-1-1 Noji Higashi,
Kusatsu, Shiga, 525-8577, Japan
{shirai, shimada}@ci.ritsumei.ac.jp

Abstract

This paper describes a choice strategy to ease user's burdens for an interactive object recognition system when the system obtains multiple object candidates as a recognition result. First, we propose several methods to display the recognition result so as to make recognition of the candidates easy and hierarchical methods to reduce candidates per choice. We verify their effectiveness by subjective tests. Next, we propose a strategy to minimize time spent for choices. We divide the time into that for displayed candidates and that for speech dialog, and formulate each time to evaluate the strategy quantitatively. Last, we compare the strategy based on choice time with a subjective strategy.

1. Introduction

Nowadays, there is a growing necessity of service robots in this aging society. For the service robots, it is one of important functions to recognize and bring user-specified objects. The object recognition system sometimes detects multiple object candidates including mistaken candidates because scenes for storage spaces of the objects such as a refrigerator and a cupboard are often complex. In this case, a user can choose the desired object from the candidates via speech dialogs with the system. The user, however, feels troublesome when too many candidates are detected.

There are many researches on reduction of user's burden when executing tasks via speech dialog. Ito et al. [1] proposed a method of reducing the number of times of dialog for a help system of electrical appliances. Ueno et al. [2] proposed a method of reducing time spent for dialog for a traffic guidance system.

In addition to those burdens of dialog, we need to consider how to display the candidates to the user so as to reduce user's burdens of visual recognition.

We can reduce burdens of user's choice by decomposing a choice from large number of candidates into multiple choices from small number of candidates. A conventional menu choice in a computer software is one of examples of the hierarchical methods and the menus are easily grouped based on their functions in advance. However, for our interactive object recognition system, we cannot group the candidates in advance because we do not know the recognition result in advance.

This paper first describes methods to display and hierarchical methods so that the user can easily choose the desired object. Next, we propose a strategy to minimize time spent for the choice with the both methods.

The outline of this paper is as follows. In sec. 2, we give an overview of our interactive object recognition system. In sec. 3, we introduce the methods to display and the hierarchical methods. In sec. 4, we formulate time spent for choice for each highlight method and hierarchy. We compare the strategy based on choice time and a subjective strategy in sec 5, and give a conclusion in sec. 6.

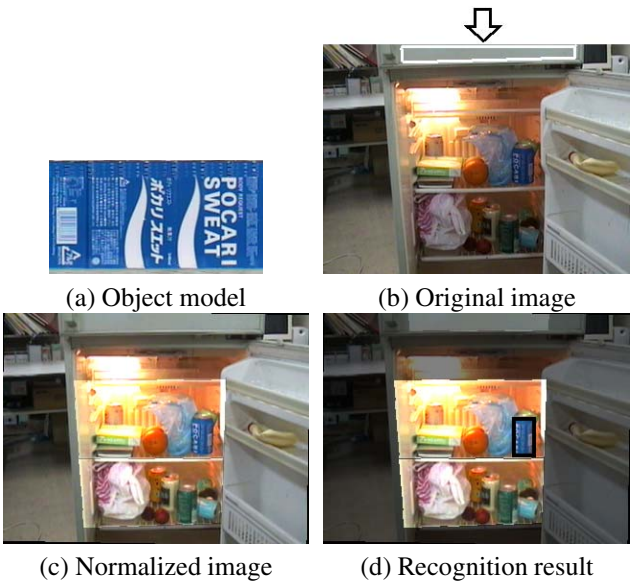
2. Interactive object recognition

We apply our interactive object recognition system to refrigerator scenes containing drinks and fruits. First, the system constructs object models in advance[5]. The model is composed of a texture image of the object (see. Figure 1(a)). Then a user asks the system to bring an object via voice.

Because illumination conditions in model construction and object recognition are different, the system estimates a current illumination condition with a reference color (average color in white box arrowed at Figure 1(b)) and transforms an original image (see Figure 1(b)) into a normalized image under the canonical condition in model construction (see Figure 1(c))[6]. Next, the system tries to recognize the object with the normalized image and displays the recognition result through display (see black box at Figure 1(d))[7]. If the system detects multiple candidates or fails in recognition, it recovers the result by user interaction[8].

We also deal with a dialog system via voice and an object manipulation system, and give details of these topics in [3] and [4] respectively.

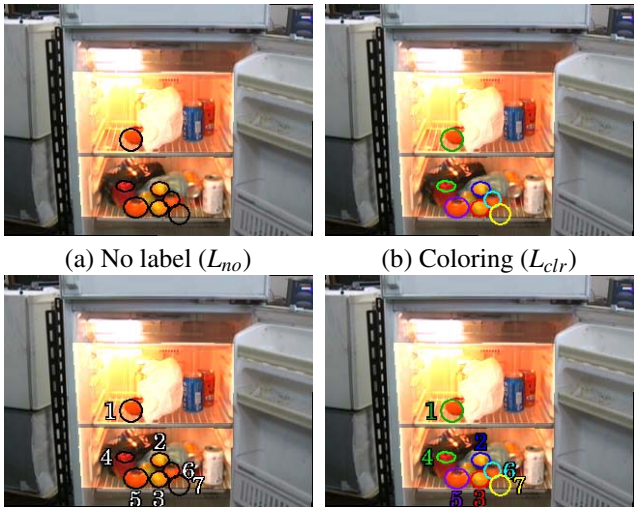
Proceedings of the 2005 International Conference on Cyberworlds (CW'05)
0-7695-2378-1/05 $20.00 © 2005 IEEE



(a) Object model (b) Original image

(c) Normalized image (d) Recognition result

Figure 1. Overview of object recognition



(a) No label (L_{no}) (b) Coloring (L_{clr})

(c) Numbering (L_{num}) (d) Numbering with color (L_{nwc})

Figure 2. Labeling to distinguish candidates

3. Methods to display and hierarchical methods

First we introduce the methods to display candidates so that the user can easily recognize them. Next we describe hierarchical methods to reduce candidates per choice when too many candidates are detected.

3.1. Methods to display candidates

We consider the following two issues for displaying candidates.

1. Labeling for specifying candidates
2. Display sequence

3.1.1. Labeling for specifying candidates We introduce 4 labeling methods below.

- L_{no}

This method is to display candidates without any labels as a recognition result as shown in Figure 2(a). The system forces the user to specify the position of the desired object in detail such as ”the second object from the left at the bottom shelf” or ”mandarin orange at the upper left of persimmon”.

- L_{clr}

This method is to display candidates with coloring as shown in Figure 2(b). The system asks the user a color name of the desired object like ”Which color region is the desired object?”, and the user answers the color name like ”Blue.” or ”Red.”.

- L_{num}

This method is to display candidates with numbering as shown in Figure 2(c). The system asks the user a number name of the desired object like ”Which number is the desired object?”, and the user answers the number name like ”Two.” or ”Three”.

- L_{nwc}

This method is to display both corresponding candidates and number with the same color. Therefore it gets somewhat easy to recognize the correspondence.

3.1.2. Display sequence The user sometimes feels difficult to judge whether the displayed region contour is really true or not when the contours are drawn on the result image. Thus, we introduce 3 methods of display sequences.

- B_{sim}

The system displays alternately the original image and the result image including all the candidates.

In this case, the ambiguity of correspondence both of the candidate and the number remains.

- B_{turn}

The system displays result images including each candidate in turn.

In this case, the correspondence of the candidate and the number is clear, but it takes longer time to display all the candidates. If the system shortens interval time T_{disp_intv}, the user sometimes overlooks a candidate in the first displaying cycle and needs to reconfirm in the second cycle. This problem often happens when elderly or the physically handicapped persons use the system.

- B_{cnf}

The system displays a candidate and to ask the user ”Is this right?”. If the user answers ”Yes.”, the system finishes confirmation. Otherwise the system displays another candidate

Proceedings of the 2005 International Conference on Cyberworlds (CW'05)
0-7695-2378-1/05 $20.00 © 2005 IEEE

sym.	label	order
D_1	L_{no}	B_{sim}
D_2	L_{clr}	B_{sim}
D_3	L_{num}	B_{sim}
D_4	L_{nwc}	B_{sim}
D_5	L_{no}	B_{turn}
D_6	L_{clr}	B_{turn}
D_7	L_{num}	B_{turn}
D_8	L_{nwc}	B_{turn}
D_9	L_{no}	B_{cnf}

val.	difficulty to choose
0	extremely difficult
1	very difficult
2	difficult
3	a little difficult
4	slightly difficult
5	neutral
6	slightly easy
7	a little easy
8	easy
9	very easy
10	extremely easy

Table 1. Methods to display and values for subjective test

and confirms to the user. The system repeats this confirmation process until the user says "Yes.".

In this case, the user pays attention only to one candidate and hence spends less concentration. Moreover, because the user's answer is limited to "Yes." or "No.", the labeling to candidates is not necessary. A demerit of this method is taking longer time.

3.1.3. Subjective test We made subjective tests for 6 persons concerning 9 methods to display as shown in Table 1. We used the scene and the candidates of Figure 2 and used 11-levels reputation values as shown in Table 1.

We had each subject see all the 9 methods to display in advance to reduce effects on results by test order of each method.

The result of the subjective test is displayed in Figure 3. Although the values are a little different among individuals, D_4 and D_7 received high values. Therefore, the system should implement some highly reputed methods and use a proper method based on user's preference and situations of recognition results.

3.2. Hierarchical methods

When too many candidates are obtained, the user feels troublesome for choosing the desired object even if the optimal method to display is used. Especially, when the system deals with fruits which have a wide variety of color and size, the system sometimes detects many candidates (see Figure 4(a)). We introduce three types of hierarchies to reduce the number of candidate per choice.

3.2.1. Hierarchy by shelf We first introduce a simple hierarchy: a hierarchy by the top shelf and the bottom shelf in the refrigerator. The system asks the user "Which shelf is the desired object on?" in the first step, and then the system asks the user to choose one on the chosen shelf in the second step. This hierarchy is effective when the number of candidates at each shelf is almost the same, and vice versa.

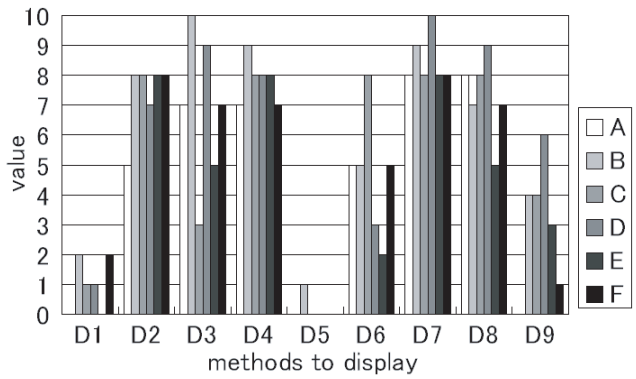


Figure 3. Result of subjective tests for methods to display

(a) All candidates (b) Representative candidates

Figure 4. Hierarchy by representative candidates

3.2.2. Hierarchy by rough group Another hierarchy uses rough groups of candidate regions at each shelf. The system first displays each connected region extracted with the object's color and asks the user "Which region includes the desired object?" at first step, and then the system has the user choose from the limited candidates belonging to the chosen region at second step.

This hierarchy works well when each candidate has a similar number of candidates. Unfortunately, one large connected candidate region is sometimes obtained when objects lie crowded. In such case, we can improve its situation by segmenting the large region into multiple small regions.

We introduce the following 2 segmentation methods.

- Dichotomy by the horizontal position (see Figure 5(a))
- Segmentation by color (see Figure 5(b))

In case that the large candidate region contains various types of fruits as shown in Figure 5, the segmentation by color has high possibility to segment the region into groups of the same types, for example greenish mandarin oranges, yellowish mandarin oranges, and misdetected objects. Then it makes possible for the user to commit the system to choose the desired object (refer to the following section).

(a) Segmentation by position (b) Segmentation by color

Figure 5. Segmentation of candidate regions

sym.	before	after
H_1	no hierarchies	hierarchy by representative candidates
H_2	no hierarchies	hierarchy by rough group with segmentation by color
H_3	hierarchy by rough group with segmentation by position	hierarchy by rough group with segmentation by color

Table 2. Items of comparison in subjective test for hierarchical methods

3.2.3. Hierarchy by representative candidate We can see that overlap of candidates is one of main causes of difficulty for user's visual recognition as shown in Figure 4(a). When the system detects overlapping candidates, the system first clusters them and chooses one representative candidate for each cluster, that is, the candidate with the best confidence as a result of object recognition (see Figure 4(b)) and then lets the others be alternative candidates. Next the system displays only the representative candidates and asks the user "Is there the desired object in these candidates?". If the user chooses from the representative candidates, the choice process is finished. Otherwise, the system displays the alternative candidates. This hierarchy is efficient in many cases because the representative candidates often include the desired object and the alternative candidates often contain misdetected candidates.

3.2.4. Subjective test We made subjective tests concerning the above hierarchical methods. We asked 6 subjects how much the choice strategy was improved relatively when 3 changes as shown in Table 2 were applied.

The result of the subjective test is displayed in Figure 6. We use 11-levels values. In this scale, values less than 5 indicate changes for the worse and those more than 5 indicate improvement.

As a result, no subjects gave worse values for the change of segmentation method. Therefore, we can see the segmentation by color is effective method.

On the other hand, two subjects gave worse values for the representative candidates and one subject did so for the rough group. These subjects said that main reason of this worse change is increase of the number of times of choices.

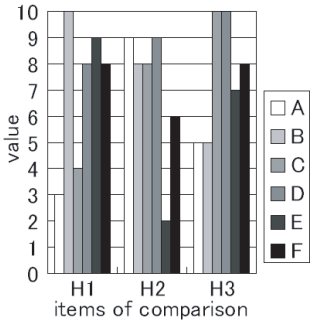


Figure 6. Result of subjective test for hierarchical methods

4. Formulation of time to choose

In this section, we formulate the time to choose the desired object by the methods to display and by hierarchical methods described in the former sections. We define the choice time as time from the first display of a recognition result to the final decision of the desired object. Here we assume that the user see an original image through display before the first display of the recognition result and confirms the approximate position of the desired object. Moreover we assume that the candidates obtained as the recognition result always include the object.

In the following sections, we exclude L_{no} because it is obviously inferior methods except for B_{cnf}, and exclude L_{nwc} because it is difficult to deal with effect both by coloring and numbering simultaneously. Thus, we use 5 methods: D_2, D_3, D_6, D_7, and D_9.

4.1. Construction of decision trees

We express the choice hierarchies as a decision tree with choice nodes. We display an example of a decision tree considering all the hierarchies in Figure 7. The top node is the choice node by shelf, and the second node is that by rough group by position or by color. The third node is that from the representative candidates, and the last choice node is that from the alternative candidates. The terminal nodes are object candidates.

Actually the system makes various decision trees by combining existence of these choice nodes, and adopts the optimal decision tree which minimizes expectation of the choice time.

$$tr^* = \arg\min_i \{\bar{T}_{tr}(tr_i)\}, \tag{1}$$

where $\bar{T}_{tr}(tr_i)$ is expectation of candidate choice time for decision tree tr_i.

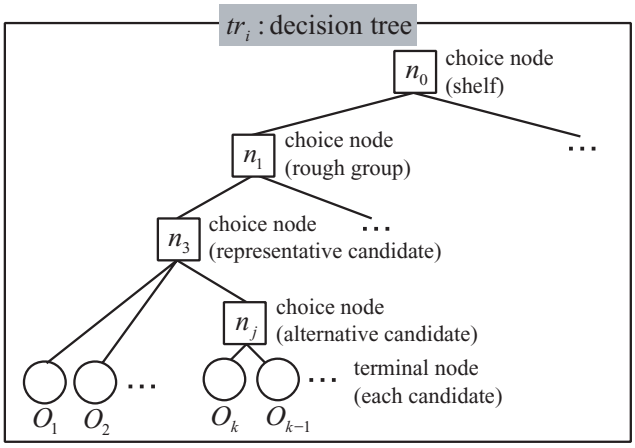


Figure 7. Decision tree for hierarchy of candidate choice

4.2. Decomposition of choice time

Because tt is difficult to directly formulate expectation of choice time $\bar{T}_{tr}(tr_i)$, we decompose $\bar{T}_{tr}(tr_i)$ into some factors. First, we decompose it into expectations for object candidates as follows.

$$\bar{T}_{tr}(tr_i) = \sum_{k=1}^{N_O} p(O_k)\bar{T}_{obj}(tr_i, O_k), \tag{2}$$

where N_O is the nubmer of candidates, $p(O_k)$ is prior probability for user's choice of candidate O_k, and $\bar{T}_{obj}(tr_i, O_k)$ is expected choice time for candidate O_k.

Moreover we decompose $\bar{T}_{obj}$ into choice times at node as follows.

$$\bar{T}_{obj}(tr_i, O_k) = \sum_{l=1}^{N_{node}} \bar{T}_{node}(n_l), \tag{3}$$

where N_{node} is the number of nodes where the user go through to reach the terminal node, n_l is l-th node from the top, and $\bar{T}_{node}(n)$ is expected choice time at node n.

In addition, choice time T_{node} at each node is expressed as the sum of the following two times.

- visual recognition time T_{rcg} spent for recognizing the desired object from displayed candidates and for recognizing the corresponding labels such as numbers or colors
- speech dialog time T_{dlg} after visual recognition

In case of B_{cnf}, because sets of user's visual recognition and speech dialog are repeated, we define T_{rcg} and T_{dlg} as the sum of repeated time respectively.

We formulate these times in the following subsections.

(a) Numbering (b) Coloring

Figure 8. Labeling for many candidates

4.3. Formulation of visual recognition time and recognition ratio

The visual recognition time T_{rcg} depends on the methods to display and the number of candidates. Moreover we take it into consideration that user sometimes misrecognizes the displayed candidates in complicated scenes. We denote a success ratio of visual recognition by p_{rcg_p} and a failure ratio by $p_{rcg\_fp}(= 1 - p_{rcg\_p})$.

- Case of B_{sim}

In case of L_{num}, when overlapping candidates are obtained as shown in Figure 8(a), user's visual recognition becomes difficult because the correspondence of thenumber and the region becomes ambiguous. Therefore, for many candidates, the visual recognition time T_{rcg} becomes longer and the visual recognition ratio p_{rcg_p} becomes worse. For simplicity, we assume that those depend on the number of candidates m, and determine relations $\bar{T}_{rcg} = T_{rcg\_ol}(m)$ and $p_{rcg\_p} = p_{rcg\_ol\_p}(m)$ by experiments.

In case of L_{clr}, the correspondence is clear even if overlapping candidates are obtained as shown in Figure 8(b). Therefore we assume that the expected time of visual recognition $\bar{T}_{rcg}$ does not depend on the number of candidates. Then we assume $\bar{T}_{rcg}$ is constant and is equal to basic visual recognition time T_{rcg_base}.

- Case of B_{turn}

If the desired object is i-th candidate, time taking to display it is

$$T_{rcg} = iT_{disp\_intv}., \tag{4}$$

where T_{disp_intv} is interval time of changing displayed candidates. Here, to prevent the user from overlooking in the first cycle, we set T_{disp_intv} as the same value of basic visual recognition time T_{rcg_base}. Assuming that the prior probability $p(O_k)$ of candidate O_k is the same, expected visual recognition time $\bar{T}_{rcg}$ for m candidates is express as:

$$\bar{T}_{rcg} = \frac{m+1}{2} T_{rcg\_base}. \tag{5}$$

- Case of B_{cnf}

methods	$\bar{T}_{rcg}$	p_{rcg_p}
D_2	T_{rcg_base}	1
D_3	$T_{rcg\_ol}(m)$	$p_{rcg\_ol\_p}(m)$
D_6	$\frac{m+1}{2}T_{rcg\_base}$	1
D_7	$\frac{m+1}{2}T_{rcg\_base}$	1
D_9	$\frac{m+1}{2}T_{rcg\_base}$	1

Table 3. Time spent for recognizing displayed candidates and recognition ratio

As mentioned before, visual recognition time is the sum of time spent for each candidate, and then it becomes the same as case of B_{turn}.

We summarize T_{rcg}, p_{rcg_p} for each method to display in Table 3.

4.4. Formulation of speech dialog time considering voice recognition ratio

The speech dialog time T_{dlg} depends on the following 2 times. The first is time T_{dlg_base} spent for the user's answer such as color names and number names, voice recognition process time, and time spent for the system's answer. . The second is time T_{dlg_cnf} spent for confirmation of the chosen object, in which the user answers "Yes." or "No.". For simplicity, we assume that T_{dlg_base} and T_{dlg_cnf} do not depend on the number of candidates and the methods to display, and determine them by experiments.

In addition, we need to consider voice recognition ratio. The voice recognition results are classified into the following 3 results and the system processes the dialog as follows (bracket indicates probability for each result).

- true positive (p_{dlg_p}): If the user chooses a terminal node, the system displays the object and confirms whether it is right. Otherwise the system moves to a next node.
- false positive (p_{dlg_fp}): The system confirms or moves to a next node in the same way as the true positive. Then the user points out the mistake, so the system asks the user to choose from candidates except for the mistaken candidate.
- negative (p_{dlg_n}): The system asks the user to choose from the same candidates again.

Those recognition ratios depends on the number of candidates m and the labeling methods L affecting user's utterance. The more the number of candidates m increases, the more the false positive probability p_{dlg_fp} increases because the number of user's utterance to be distinguished each other increases. When similar colors are used because of too many candidates as shown in Figure 8(b), the false positive may sometimes happen, for example, between "yellow green" and "green". In case of confirmation, the user says only "Yes." or "No.", so we assume that the system always succeeds in voice recognition.

Moreover we take the visual recognition ratios p_{rcg_p} into consideration together. Here we assume that visual recognition result and voice recognition result are independent each other, and define joint true positive, false positive, and negative as p_{tp}, p_{fp}, and p_n respectively. The joint true positive is mainly dominated by one case: true positive both of visual recognition and voice recognition. The joint false positive is mainly dominated by two cases: true positive of visual recognition and false positive of voice recognition, and false positive of visual recognition and true positive or false positive of voice recognition. Then we define them as follows.

$$p_{tp} = p_{rcg\_p}p_{dlg\_p} \quad (6)$$

$$p_{fp} = p_{rcg\_p}p_{dlg\_fp} + p_{rcg\_fp}(p_{dlg\_p} + p_{dlg\_fp}) \quad (7)$$

$$p_n = 1 - (p_{tp} + p_{fp}) \quad (8)$$

We formulate the speech dialog time considering the above probabilities below.

- Case of B_{sim} or B_{turn}

We first formulate expected speech dialog time $\bar{T}_{dlg}(m)$ when the user chooses from m terminal nodes. All dialog cases are divided into two cases: first is reaching true positive and finish with confirmation after several times of negatives and second is reaching false positive and moves to choice from reduced number of candidates through user's collection after several negatives. As a result, we obtain

$$\begin{aligned}\bar{T}_{dlg}(m) &= \sum_{i=1}^{\infty}(p_n)^{i-1}p_{tp}(iT_{dlg\_base} + T_{dlg\_cnf}) \\ &+ \sum_{i=1}^{\infty}(p_n)^{i-1}p_{fp}(iT_{dlg\_base} + T_{dlg\_cnf} + \bar{T}_{dlg}(m-1)) \\ &= \frac{T_{dlg\_base}}{p_{tp}+p_{fp}} + T_{dlg\_cnf} + \frac{p_{fp}}{p_{tp}+p_{fp}}\bar{T}_{dlg}(m-1) \\ T_{dlg}(1) &= T_{cnf}. \end{aligned} \quad (9)$$

Next, we formulate $\bar{T}_{dlg}(m)$ when the user chooses from nodes other than terminal nodes. In this case, the system omits confirmation when true positive or false positive, so $\bar{T}_{dlg}(m)$ is

$$\bar{T}_{dlg}(m) = \frac{T_{rcg\_base}}{p_{tp}+p_{fp}} + \frac{p_{fp}}{p_{tp}+p_{fp}}\{\bar{T}_{dlg}(m-1) + T_{dlg\_cnf}\}. \quad (10)$$

- Caes of B_{cnf}

The system repeats m times of confirmation processes in the worst case, therefore expected speech dialog time is

$$\bar{T}_{dlg}(m) = \frac{m+1}{2}T_{dlg\_cnf}. \quad (11)$$

4.5. User's commitment of choice

In the above sections, we assumed that the user always makes a choice at each node until reaching a terminal node.

However, when all the candidates at a node consist of the same type of objects, the user can transfer the choice to the system like "Any one is all right." and saves time by omitting the following choices.

There are several cases where the user transfers the choices. For example, if all the candidates obtained as a recognition result consist of the same type of objects, the user can transfer the choice from the first. Also, in cases of choice by shelf, by rough group, and by representative candidates, the user can transfer the choice in the same way.

In general, the user tends to put the same type of objects together at almost the same position, so a rough group by position sometimes consists of the same type of objects. A rough group, however, sometimes consists of mixed type of objects when the numbers of each type of objects are different or when the objects are not places together. On the other hand, the rough group by color more often consists of the same type of objects even if the objects are not places together because the same type of objects probably have the almost the same color respectively. Therefore the user's transference more often happens in case of the rough group by color than in case of the other hierarchies.

5. Comparison of strategy based on choice time and subjective strategy

For simplicity, we limit decision trees into the following two types: single-layer decision tree in which the user directly chooses from object candidates and double-layer decision tree in which the user chooses from rough groups in the first step and in which chooses from limited object candidates in the second step. In general, as the number of candidates increases, the user prefers the double-layer decision tree. Therefore, we compare the number in which the system switches the single-layer decision tree to the double-layer one with the number in which the user does so.

5.1. Parameterization for choice time

We need to determine the paramters which are used to formulate the choice time. The first is the interval time T_{disp_intv}, which is assumed to be equal to the basic visual recognition time T_{rcg_base}. We ask subjects to see object candidates for various interval times and then determines the minimum time without overlook as T_{disp_intv}. The second is the visual recognition time $T_{rcg\_ol}(m)$ for D_2. We ask the subjects to see several scenes and choose the desired object for each scene. Then we store the number of all the candidates m and visual recognition time T_{rcg_ol}, and we obtain the linear relation between them by approximation. In addition, we determine the visual recognition ratio p_{rcg_p} by experiments. Last, we determine the voice recognition ratios by experiments using voice recognition software "Julius"[9].

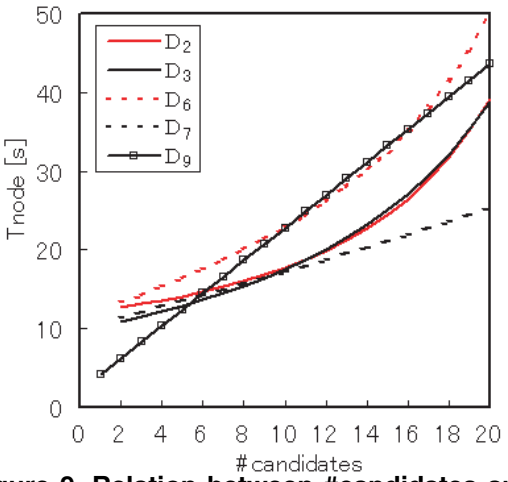


Figure 9. Relation between #candidates and expected choice time

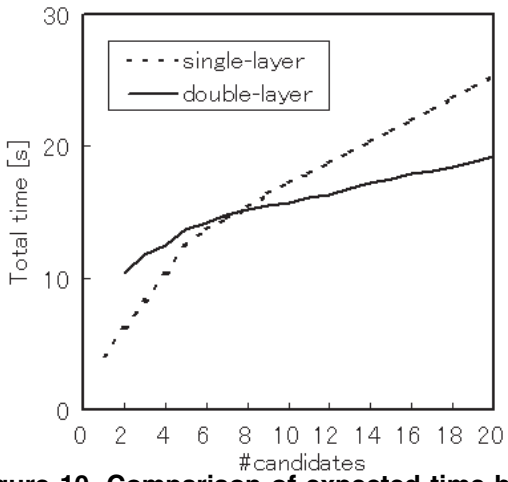


Figure 10. Comparison of expected time between single-layer decision tree and double-layer one

5.2. Simulation of choice time

We display relations of the number of candidates and the candidate choice time for the single-layer decision tree in Figure 9. We can see that the confirmation one by one D_{cnf_no} is effective for small m and displaying in turn with the numbering D_{turn_num} is effective for large m. In actual, the system adopts the method to display which minimizes the choice time for each m.

Next, we display relations of the number of candidates and the choice time for the single-layer decision tree and the double-layer one in Figure 10. In this example, the number $m = 7$ is threshold whether the system adopts the single-layer decision tree or the double-layer one.

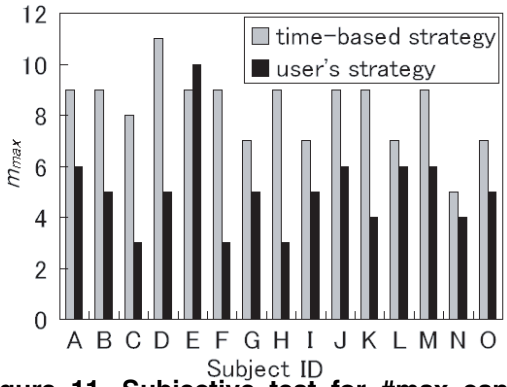


Figure 11. Subjective test for #max candidates for single-layer choice

5.3. Comparison with subjective strategy

We asked 15 subjects to determine the max number of candidates (let it be m_{max}) in case of adopting the single-layer decision tree. We display m_{max} for the strategy based on choice time and that for the subjective strategy in Figure 11. We can see all the subjects sets m_{max} less than or equal to that based on choice time except subject E. Therefore we need to consider not only the choice time but also mental burdens like powers of concentration at each choice to reduce total user's burdens.

6. Conclusion

We described a strategy of choice to ease user's burdens for an interactive object recognition system when the system obtains multiple candidates as a recognition result. First, we proposed several methods to display the recognition result so as to recognize candidates easily and hierarchical methods to reduce candidates per choice, and verified their effectiveness by subjective tests. Next, we propose a strategy to minimize time spent for choices. We divided the time spent for the choice into visual recognition time and speech dialog time, and formulated each time to repute the strategy quantitatively. Last, we compared the strategy based on choice time with the subjective strategy and discussed the necessity of considering the user's mental burdens such as powers of concentration.

Future works are as follows.

- Subjective tests for general users
- Choice strategy considering user's mental burdens such as power of concentration

References

[1] R. Ito, K. Komatani, and T. Kawahara, "Spoken Dialogue Help System for Electrical Appliances Using Knowledge and Structure of Their Manuals", J. of Information Processing Society of Japan, Vol.43, No.7, pp.2147-2154, 2002.

[2] S. Ueno, K. Komatani, T. Kawahara, and H.G. Okuno, "Experimental Evaluation and User Modeling for Kyoto City Bus Information System", Proc. of the 17th Annual Conf. of the Japanese Society for Artificial Intelligence, 2C2-03, 2003.

[3] M. Takizawa, Y. Makihara, N. Shimada, J. Miura, and Y. Shirai, "A Service Robot with Interactive Vision -Object Recognition Using Dialog with User", Proc. of 1st Int. Workshop on Language Understanding and Agents for Real World Interaction, pp. 16-23, Jul. 2003.

[4] J. Miura, Y. Shirai, N. Shimada, Y. Makihara, M. Takizawa, and Y. Yano, "Development of a Personal Service Robot with User-Friendly Interfaces", Proc. of the 4th Int. Conf. on Field and Service Robotics, pp. 293-298, Jul. 2003.

[5] Y. Makihara, M. Takizawa, Y. Shirai, J. Miura, and N. Shimada, "Object Registration Supported by User Interaction for Service Robots", Proc. of the 19th Annual Conf. of the Robotics Society of Japan, CD-ROM, 2001 (in Japanese).

[6] Y. Makihara, Y. Shirai, and N. Shimada, "Online Learning of Color Transformation for Interactive Object Recognition under Various Lighting Conditions", Proc. of the 17th Int. Conf. on Pattern Recognition, Vol. 3, pp. 161-164, Cambridge, UK, Aug. 2004.

[7] Y. Makihara, M. Takizawa, Y. Shirai, J. Miura, and N. Shimada, "Object Recognition Supported by User Interaction for Service Robots", Proc. of 16th Int. Conf. on Pattern Recognition, Vol. 3, pp. 561-564, Quebec, Canada, Aug. 2002.

[8] Y. Makihara, M. Takizawa, Y. Shirai, J. Miura, and N. Shimada, "Object Recognition Supported by User Interaction for Service Robots", Proc. of 5th Asian Conf. on Computer Vision, Vol. 2, pp. 719-724, Melbourne, Australia, Jan. 2002.

[9] "Julius -an Open-Source Large Vocabulary CSR Engine-", http://julius.sourceforge.jp/en/julius.html