Paper:

Cooking Procedure Recognition and Support by Ubiquitous Sensors

Sho Murakami, Takuo Suzuki, Akira Tokumasu, and Yasushi Nakauchi

Graduate School of Systems and Information Engineering, University of Tsukuba
Tsukuba, Ibaraki 305-8573, Japan
E-mail: {murakami, takuo, tokumasu, nakauchi}@hri.iit.tsukuba.ac.jp
[Received January 15, 2009; accepted July 13, 2009]

This paper proposes cooking support using ubiquitous sensors. We developed a machine learning algorithm that recognizes cooking procedures by taking into account widely varying sensor information and user behavior. To provide appropriate instructions to users, we developed a Markov-model-based behavior prediction algorithm. Using these algorithms, we developed cooking support automatically displaying cooking instruction videos based on user progress. Experiments and experimental results confirmed the feasibility of our proposed cooking support.

Keywords: intelligent environment, ubiquitous sensors, cooking support, machine learning

1. Introduction

Recent developments in information technology are computerizing and networking electric household appliances. If environments in which users work could recognize user activities indirectly through the use of sensors, novel services suiting user activities could be realized. An important factor here is the recognition of user behavior by ubiquitous sensors - an idea initially proposed by Weiser as ubiquitous computing [1] and that has emerged in such forms as the Aware Home [2], Robotic Room [3], and Yukari projects [4].

Cooking is one of the most complex of all household tasks. The types of dishes used vary widely and are highly specialized. Cooking must take into account nutrition, preferences, and climate seasons. Users cooking a dish for the first time usually follow cookbook instructions. Cookbooks, however, tend to take up kitchen space and are difficult to reference while cooking is in progress. Having kitchen-embedded sensors recognize cooking progress, for example, could provide instructions to users on what to do next based on cooking progress.

The cooking support we developed recognizes user activity by sensing the movement of ingredients and kitchen utensils labeled with radiofrequency identification (RFID) tags. It determines the next ingredients or kitchen utensils to be used based on previous user behavior, and recommends users to use them [5]. Using only RFID tags for sensors, however, prevents the recognition of cutting procedures independent of RFID tag movement or cooking procedures requiring that food be heated. To comprehensively recognize cooking procedures, Yamakata et al. realized a system that recognizes ingredients and whether preparation involves cutting or peeled on a cutting board through the use of CCD and infrared cameras [6]. They developed recipe generation that automatically divides cooking videos into video clips and associates them with corresponding recipe text [7]. Shiio et al. developed a "kitchen of the future" that displays cooking procedures on multiple screens that users control using a foot switch [8]. They developed cooking support using "happy cooking" [9] that optimizes cooking procedures even when more than one dish is cooked simultaneously. The "kitchen of the future" does not recognize cooking procedures, however, so users must request information by pressing the foot switch each time they need such information.

No system currently recognizes and supports the entire cooking process, so we propose cooking support recognizing user activities using sensors and supporting users based on their progress through a cooking procedure. Ubiquitous sensors are embedded in a kitchen to recognize cooking progress. We use RFID tags to sense the movement of ingredients and cooking utensils, CCD and infrared cameras for detecting ingredient color and temperature distribution on a cutting board, and laser range finders for determine user stance. To recognize cooking procedures via sensor information, we use a machine learning algorithm that automatically generates recognition rules. Complex cooking procedures are difficult to recognize accurately, however, so we propose an original machine learning algorithm that estimates multiple recognition candidates and probabilities.

Information on the cooking procedure currently in progress or to be done next should be presented to support users appropriately. For users peeling carrots, for example, the carrot-peeling instructions corresponding to the procedure should be presented. For users removing a carrot from a pantry, however, a carrot-washing instruction corresponding to the next procedure should be presented. Peeling a carrot takes time, so the instruction should be current ongoing work, whereas taking a carrot out of a pantry is quick, so the instruction should be that to be

Table 1. Verbs extracted from cookbook.

recognized procedures	synonyms	percentage in all verbs(%)
wash	wash	1.95
cut	cut	10.73
	peel	1.83
	make a slits	0.74
	cut away	0.81
	remove	1.17
	trim	1.19
	mince	0.17
	flatten	0.04
	quarter	0.09
	chop up	0.64
dress and mix	mix	0.83
	combine	0.64
	stir	3.12
	dress	0.30
	put on	0.04
	add	2.72
	insert	2.55
drain	drain	3.06
	dry	0.45
put on a pot	put on a pot	1.53
	heat	2.53
add to a pot	add to a pot	7.80
	insert	4.46
boil	boil	4.10
	poach	2.61
	set aflame	0.85
saute	saute	3.27
	fry	1.42
	brown	0.09
deep-fry	deep-fry	0.85
steam	steam	0.47
serve	serve	3.10
	garnishe	1.06
	sprinkle	0.68
	pour	0.49
total		68.38

done next. To support users appropriately, we propose a Markov-model-based algorithm predicting procedures and probabilities of what users should execute next. This automatically provides appropriate information to users when predicted probability is high. User instruction candidates can then be shown for users to choose from when predicted probability is low.

We propose cooking support using the above cooking procedure recognition and prediction algorithms. Predicted cooking procedure instructions are displayed one after another together with appropriate video clips and voice instructions. We conducted experiments to confirm the efficacy of our proposal.

2. Cooking Procedures

To support cooking procedures by providing appropriate information based on the cooking progress, we must recognize procedures at an appropriate level.

We extracted verbs used in 253 recipes in a cookbook [10] containing 200 procedures and 5,000 expressions and merged cooking into the procedures listed in **Table 1**.

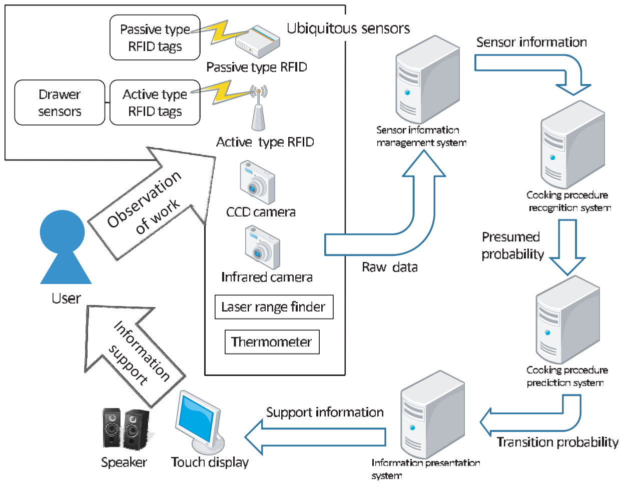

Fig. 1. Configuration of the cooking support system.

Procedures "wash," "cut," "dress and mix," and "drain" are used in ingredient preparation. Because the type of cutting done depends on the type of ingredient, we divided "cut" into "cut XX," where XX is the ingredient to be cut. Procedures "put on a pot," "add to a pot," "boil," "saute," "deep-fry," and "steam" are used in cooking. Procedure "dish-serve" was used. These procedures cover 68% of all verbs in the cookbook and almost all procedures needed in cooking [1].

Procedure "take an ingredient" not used in the cookbook is, however, important in recognizing cooking progress, so we included it as a recognition target. Because users may be standing still or moving around without doing anything in particular, we added "moving" as a recognition target for these cases.

3. Cooking Support

The configuration of our proposed cooking support, shown in **Fig. 1**, consists of 4 subsystems – 1) sensor information management 2) cooking procedure recognition 3) cooking procedure prediction 4) information presentation – explained below.

3.1. Sensor Information Management

Sensor information management receives information from ubiquitous embedded kitchen sensors, compresses and converts this information into a data format used easily by recognition, and stores them in a database. To recognize targeted cooking procedures described in Section 2, we use RFID tags to sense the movement of ingredients and cooking utensils, CCD and infrared cameras for detecting the color and temperature distribution of ingredients on a cutting board, and laser range finders for recognizing user locations. Sensors use different sam-

1. Procedures we did not target were "cool" and "spare some time," which do not accompany any explicit operation. "Turn" and "allow flavor to blend," which is a part of "saute," is already included in recognition targets.

pling times, so we aligned them as Time when they were stored in the database, as detailed in Section 4.1.

3.2. Cooking Procedure Recognition

Cooking procedure recognition recognizes procedures a user is currently executing based on sensor information management. We used machine learning to predict multiple procedure candidates with probability.

Neither binary classifiers such as support vector machine (SVM) classifiers nor those whose recognition rules are black boxes, i.e., neural networks, are appropriate for getting multiple procedure candidates. Bayesian networks require a system developer to design network configurations, so we used decision tree algorithm C5.0 [11], which uses entropy in information theory, as follows: A decision tree generated by C5.0 consists of nodes and leaves (classes denoting recognition results). The recognition rule is represented as sets of nodes from roots to leaves. C5.0 recreates new decision trees by checking false recognitions of learning instances. This does not necessarily improve decision tree quality linearly, however, so we use several decision trees generated by such "boosting." Recognition results are calculated as the average of multiple recognition results derived by multiple decision trees.

In calculating the probability of recognition results, we define recognition rules as $r_1, r_2, \dots, r_l$, and a set of rules as R. The number of rules is the same as the number of leaves on the decision tree. We define learning instance as $t1, t2, \dots, tm$, and a set of learning instances as T. We define proposition $\mathrm{Match}(r_j, t_i)$, where learning instance ti satisfies all decision nodes in rule rj. The training data set that satisfy rule rj is as follows:

$$T_j = \{t_i | \mathrm{Match}(r_j, t_i)\} \quad (i \in \forall T, j \in \forall R). \quad . \quad (1)$$

In C5.0, the leaf (class that represents recognition result) must be unique. When a certain leaf node has four "cut lettuce" instances and six "cut cabbage" instances, for example, C5.0 makes the leaf node a "cut cabbage" leaf because it has more instances. As the result, the system always determines "cut cabbage" even if a user is cutting lettuce which may occur at 40%. So we improve C5.0 to maintain multiple leaves with probability.

We define probability E_{jk} of class c_k in recognition rule r_j as follows:

$$E_{jk} = \frac{\mathrm{freq}(c_k, T_j)}{|T_j|} \quad (j \in \forall R, k \in \forall C) \quad . \; . \; . \; . \quad (2)$$

$|Set|$ is the number of elements in Set. $c_1, c_2, \dots, c_n$ denote classes to be recognized, and C denotes their set. $\mathrm{freq}(c_k, T_j)$ is the number of elements belonging to class c_k in learning instances T_j.

$|T_j|$ is the number of training data that satisfies recognition rule r_j. The recognition of cooking procedure with a small number of $|T_i|$ is not reliable, so we prune the decision tree so recognition is supported by a sufficient number of learning instances. The pruning algorithm is as follows:

When the number of learning instance satisfying recognition rule r_j is less than α, the furthest node from the root is pruned and a new leaf is created, probabilities E_{jk} are recalculated for the new leaf, and this process is repeated until the number of learning instance exceeds α. If the number of learning instance does not exceed α even when β nodes are deleted, probabilities E_{jk} are set to 0 because the number of similar events occurring in the past is not sufficient. With C5.0, nodes are created from the root to leaves based on entropy, so this way of deleting a node from leaves is considered appropriate.

The pruning algorithm and probability E_{jk} calculation will be executed offline using learning instances beforehand. Cooking procedure recognition finds appropriate rule r_j in the decision tree by using sensor information from the sensor information management. The recognition result is probabilities denoted as E_{jk}.

This operation is done for multiple decision trees generated by boosting, and E_{jk} are averaged for each class k. Averaged probabilities $\bar{E}_k$ for each class k are passed to cooking procedure prediction.

3.3. Cooking Procedure Prediction

Cooking procedure prediction predicts procedures users should execute next using recognition results from cooking procedure recognition. We predict procedures users should execute next with probability using the Markov model.

The Markov chain, a discrete stochastic process, assumes a Markov property, and the model based on the Markov chain is a Markov model, which is expressed by transition probability matrix M, where m_{ij} denotes the transition probability from state i to state j. Because the numbers of learning instances may be insufficient and cooking procedures not influenced from past procedures, we use a simple Markov model.

Suppose that procedures "cut cabbage" and "cut lettuce" are recognized at the same probability by cooking procedure recognition. Since these procedures are similar – both ingredients are green and similar in shape – both probabilities should be considered equally when predicting the next procedures users should take. We use transition probability P_j, which takes multiple probabilities simultaneously, as follows:

$$P_j = \sum_i \bar{E}_i \cdot m_{ij} \quad (i, j \in \forall C). \quad . \; . \; . \; . \; . \; . \quad (3)$$

In making the Markov model, a learning instance is created as a time series of classes at sampling time $Time$. The Start class representing the start of cooking is inserted at the top of the learning instance. Transition probability matrix M is created by calculating transitions between classes. Class "moving" is omitted here because it occurs so often and does not of itself represent a cooking procedure.

3.4. Information Presentation

Information presentation predicts procedures users should execute next by using transition probabilities. The system provides appropriate information to users automatically when predicted probability is high, but shows the candidates of instructions for users to choose when predicted probability is low.

Take the example of "take a dressing" which is short in time was conducted after "cut cabbage" which usually takes long in time. The transition probability that loops back from "cut cabbage" to "cut cabbage" will be high since it usually lasts long. The transition probability from "cut cabbage" to "take a dressing" will be low. However, the transition probability from "cut cabbage" to "take a dressing" will be increased since the same status ("cut cabbage") lasts long. In this way, the probability will be accumulated in time and will exceed certain threshold at the appropriate transition timing.

We made the cooking support to accumulate transition probability P_j as S_j, and procedure S_j with high probabilities are considered as candidates to support a user.

$$S_j(t) = S_j(t - Time) + P_j \quad (j \in \forall C). \quad . \quad . \quad . \quad (4)$$

Information is presented using S_i. When a candidate S_i satisfies the following equation, information presentation automatically plays cooking information which concerns the procedure S_i to a user:

$$S_i > \gamma \quad \wedge \quad S_i > \delta \cdot S_j \quad (i \neq j, \quad i, j \in \forall C). \qquad (5)$$

γ is a threshold for automatically helping users. δ is a threshold for subsequent candidates to be considered. If no S_i satisfies the equation, all cooking procedure items that satisfies $S_i > \varepsilon$ will be displayed as buttons for a user to choose.

S_j increases monotonically with time, even if P_j has a low transition probability. For the monotonic increase of S_j, once the cooking procedure is presented to a user (either by automatic or by the button selection), we made $S_j = 0$. For the low transition probability of P_j, we employed the forgetting coefficients ζ:

$$S_j(t) = S_j(t - Time) - \zeta(P_j < \eta, S_j(t - Time) \geq 0). \quad (6)$$

η is a threshold for judging whether transition probability P_j is small enough.

Since used ingredients and cooking utensils and sequences of cooking procedures vary with the recipe, cooking procedure recognition data and information presentation are prepared for each recipe. Users select recipes before starting cooking.

4. Implementation

Kitchens generally consist of three areas - a sink, a counter, and a range (heating counter). In Japan, kitchens are arranged with the sink, counter, and range aligned in a row, as is done in our experiments, with ubiquitous sensors embedded in this arrangement. We implemented proposed cooking support consisting of sensor information management, cooking procedure recognition, cooking procedure prediction, and information presentation. Sensor information management was implemented using Java and other subsystems using $C\#$.

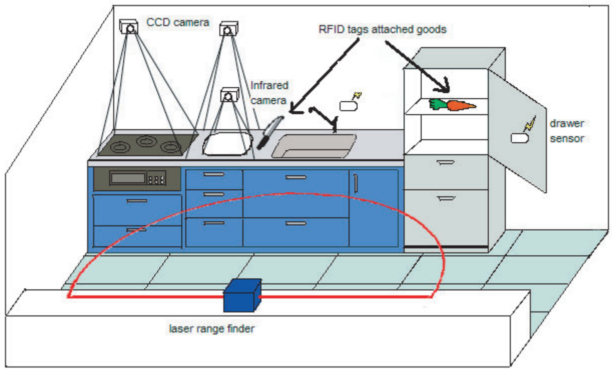

Fig. 2. Embedded ubiquitous kitchen sensors.

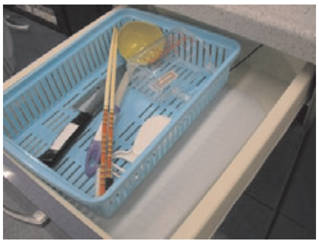

Fig. 3. RFID antenna in a drawer and RFID tags on cooking utensils.

Fig. 4. RFID antenna in a pantry and RFID tags on ingredients.

4.1. Information Management

Sensor information management receives information from embedded ubiquitous sensors as shown in **Fig. 2**, compressing and converting information and accumulating it in a database. Information is accumulated in the database at sampling time $Time = 1(sec)$ as follows:

4.1.1. Sensors for Ingredients and Cooking Utensils

Information on ingredient and cooking utensil movement is very informative for recognizing user cooking procedures, which is why we put passive RFID tags on kitchen ingredients and utensils. RFID tags are already in wide use at major retail enterprises such as Wal-Mart for stock management, and are expected to replace bar codes in the near future. We installed antennas in the ingredient pantry, in the utensil drawer, and beneath the kitchen counter to comprehensively observe ingredient and utensil movement, as shown in **Figs. 3** and **4**.

Antennas detecting RFID tags and storing such information in the database observe changes in RFID tags, e.g., potatoes disappearing from the pantry, and record changes with information when it is observed and how long that status lasts. Knowing items that have disappeared or were moved most recently is useful in recognizing current user procedures.

4.1.2. Drawer Sensors

Ingredient and utensil drawer status is useful in recognizing cooking procedures, so we put magnet sensors on

Fig. 5. Magnet sensor on a drawer.

Fig. 6. Limit switch sensor on a faucet.

kitchen doors and drawers as shown in **Fig. 5**. To detect water use, we put a limit switch on the faucet and transmitted such information using active RFID tags as shown in **Fig. 6**.

4.1.3. User Position Sensors

User location and cooking procedures are correlated, so we used two laser range sensors [2] to detect user locations, Which we classified into "in front of pantry," "in front of sink," "in front of counter," "in front of range," and "others" for areas away from the I-shaped kitchen. Current and previous user locations and elapsed times among them are stored in the database.

4.1.4. Kitchen Counter Sensors

Cooking preparation procedures are conducted at the counter. Knowing user hand locations with relation to the counter is useful for knowing whether the user is working at the counter and knowing colors of ingredients to determine what is being cut on the cutting board.

The cutting board, hands, and ingredients on the counter simultaneously may make it difficult to detect the color of ingredients using a CCD camera alone. These items all have different temperature distribution, however, so we installed CCD cameras to acquire color information and infrared camera [3] to determine the temperature distribution above the kitchen counter, thus obtaining the size of the hand photographed and the color of ingredients on the cutting board.

To obtain hand size, the hand is assumed to be flesh-colored and to have a surface temperature of $\geq 26°$C. The flesh-colored area is detected as follows:

$$\begin{cases} 0.4 < R' < 0.5 \\ 0.25 < G' < 0.4 \\ G' < R'. \end{cases} \qquad \left(\begin{array}{l} R' = \frac{R}{R+G+B} \\ G' = \frac{G}{R+G+B} \end{array} \right) \quad . \quad . \quad (7)$$

Flesh-colored areas having a temperature is $\geq 26°$C are recognized as a hand and their size is accumulated in the database.

Ingredient colors are detected as follows: We assume that ingredients are 16°C because they are cooled in a refrigerator [4]. Their brightness is also expected to change due to shadows, e.g., from a hand. We use a color space of hue, saturation, and brightness (HSV) [12] of accumulated in the database.

2. HOKUYO URG-04LX
3. IRISYS Universal Thermal Imager TypeIRI 1011
4. In experiments, we used ingredients kept in a refrigerator.

4.1.5. Range Sensor

Pans on the range are heated during cooking. Pans reach a high temperature and users stir ingredients. Types of heating procedures such as boiling and deep-frying use different cooking temperatures, so we used an induction heating (IH) system for cooking [5] to measure the temperature of the pan. We used a CCD camera over the range to determine if stirring was being done. The pan temperature and hand size observed by the CCD camera are stored in the database.

4.2. Cooking Procedure Recognition

Cooking procedure recognition creates a decision tree offline. When a cooking procedure is recognized, cooking procedure recognition puts sensor information in the decision tree to get results once each second. Learning instances consisting of 103 sensor information attributes were prepared by putting classes (teacher data) in the database. The decision tree was generated using C5.0 [11] libraries, and boosts numbered 10. Parameters for the pruning algorithm were $\alpha = 3$ and $\beta = 3$.

4.3. Cooking Procedure Prediction

Cooking procedure prediction was implemented using the learning instance accumulated by sensor information management constructing a transition probability matrix based on the Markov model.

From observing decision trees, we found that classification nodes for user position sensors and RFID tag sensors come from root to leaves, meaning that user position sensors provide the most information for recognizing cooking procedures, followed by RFID tag information on ingredient and cooking utensil use.

4.4. Information Presentation

We developed information presentation using a touch display and a speaker. Parameters for determining the timing of support were $\gamma = 0.8$, $\delta = 1.6$, and $\varepsilon = 0.05$. Parameters for forgetting factors were $\zeta = 0.2$ and $\eta = 0.0001$.

Video instructions on "how to cut an onion" presented automatically to users cutting onions are shown in **Fig. 7**. After users finish cutting, candidates "take a carrot" and "take a potato" are presented as buttons for users to choose between as shown in **Fig. 8**. If a user push a button for location of the ingredient, the picture of system kitchen will be displayed and the location will be indicated as a dot on it.

5. Experiments and Discussion

5.1. Cooking Procedure Recognition

To evaluate the accuracy of cooking procedure recognition in Section 4.2, we had subjects cook several dishes.

5. Matsushita Electric Industrial Co., Ltd. (laboratory specification) KZ-VSW32B.

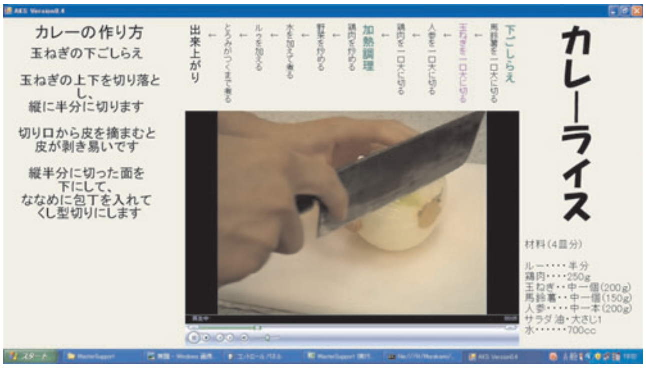

Fig. 7. Automatically displayed text and video instructions.

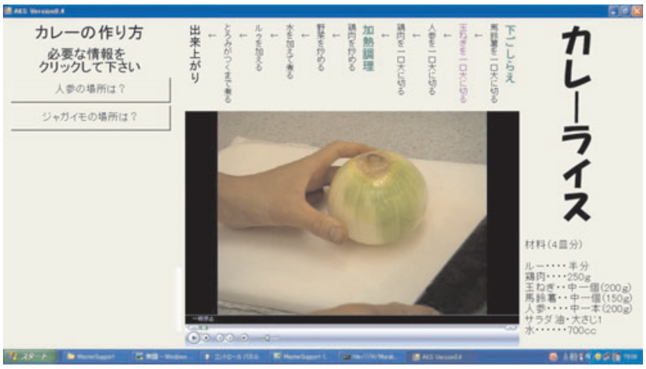

Fig. 8. Dynamically generated buttons displayed to users.

To measure recognition accuracy for all cooking procedures in **Table 1**, we asked 10 subjects to prepare "curry" and "salad" and five to prepare "deep-fried chicken" and "pot-steamed hotchpotch."

To calculate recognition accuracy, we conducted cross-validation by treating cooking procedures by one subject as one data item. We used the data for the other 9 subjects for learning because 5 subjects' data when we conduct 2-fold cross-validation is not sufficient to make the system learn. Recognition accuracy for individual dishes is shown in **Figs. 9 - 12**. Recognition accuracy is shown by two bars. Bars at left denote the precision of classes with the highest probabilities. Bars at right denote the accumulation of precision including all candidates whose probability is 10% or more. In fact, 5 candidates were selected at most.

As shown in figures, the most of the recognition accuracies of the first candidate are more than 80% and very high. From the results, we confirmed the cooking procedure recognition system is using C5.0 functions effectively. However, some of the recognition accuracy are observed as 60% $\sim$ 70%. A right bars are for confirming whether the correct answer is included by using the procedures whose probability is more than 10%. From the result, we confirmed most procedures were predicted at the probability of 95% $\sim$ 100%.

As the figures show, the most of the recognition accuracies of the first candidate are more than 80% and very high. From the results, we confirmed the cooking procedure recognition using C5.0 worked efficiently. Some recognition accuracy was 60% $\sim$ 70%. Bars at right

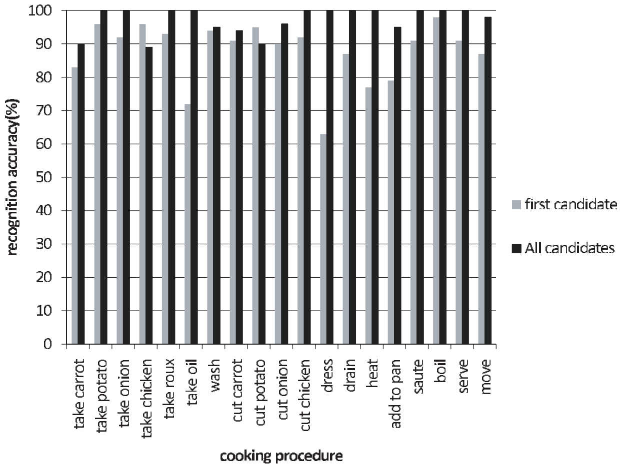

Fig. 9. Recognition accuracy for curry.

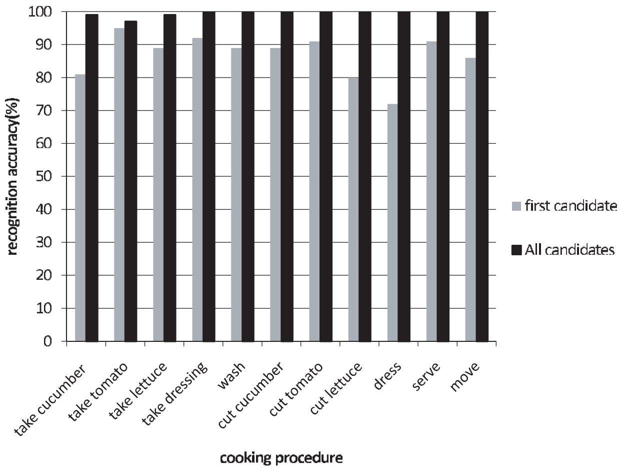

Fig. 10. Recognition accuracy for green salad.

are for confirming whether the correct answers are included by using the procedures whose probability is more than 10%. As the figures show, the results confirmed that most procedures were predicted at a probability of 95% $\sim$ 100%.

Cooking procedures with low recognition accuracy are typically finished quickly, indicating that they have very few learning instances. For this reason, we believe the system could be improved by increasing the number of learning instances.

We observed that procedures such as "take soy sauce" and "dressing" were conducted simultaneously by some subjects, so individual procedures were not recognized appropriately, which is why recognition accuracy for "dressing" was rather low.

5.2. Cooking Procedure Support

We conducted experiments with subjects to confirm the efficiency of the cooking support. We asked ten subjects other than the subjects we used at the previous experiments to cook "curry" – the most complex of the four recipes. For cooking procedure recognition and cooking procedure prediction, data used was obtained in previous experiments (5.1).

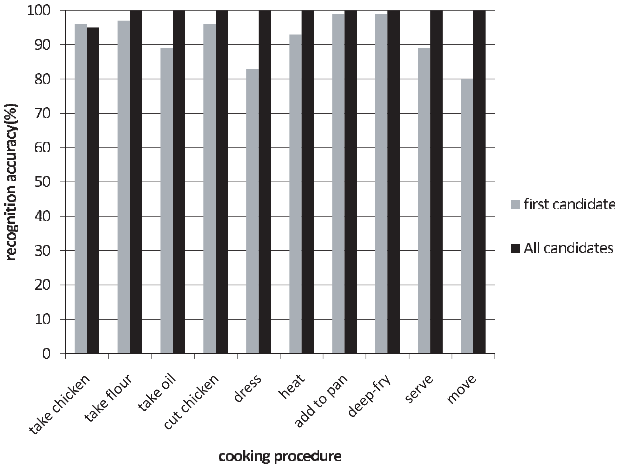

Fig. 11. Recognition accuracy (Recognition accuracy for fried chicken).

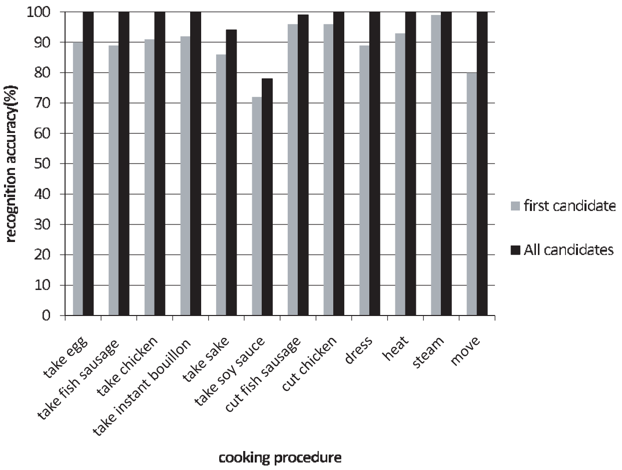

Fig. 12. Recognition accuracy for steamed egg custard.

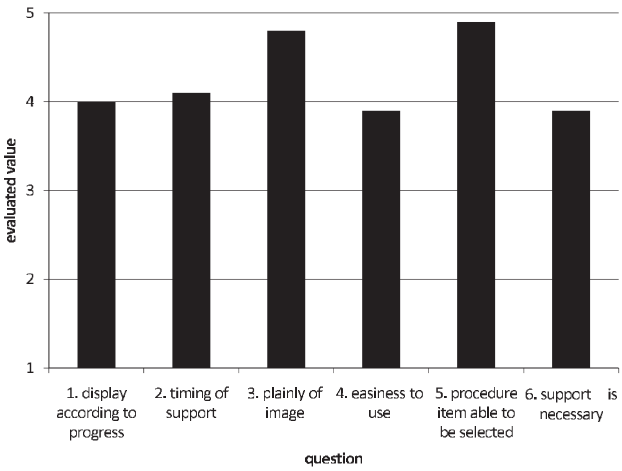

Fig. 13. Evaluation of our proposal.

We asked subjects to make evaluations by answering questionnaires after cooking. The results of questionnaires are shown in **Fig. 13**.

Evaluation of whether videos automatically shown to users were appropriate was 4.0, therefore, the system provided appropriate information based on cooking progress.

Evaluation of whether timing of videos shown to users was appropriate was 4.1 and satisfactory. Some subjects answered showing videos was "early" or "a little early" but none answered that they were a "little slow" or "slow." Since information candidates are displayed to users, subjects who need information could select it by pressing a button, so none felt that information presentation was slow because when they needed it, they simply choose it.

Evaluation of image and the voice support was very high at 4.8. Voice support was favored because it did not interrupt cooking procedures.

Evaluation of the touch display was 3.9. Some respondents felt that response time was slow because of wet fingers and the need to consider meal preparation sanitation. We are thus considering using voice recognition in future implementation.

Evaluation of support items displayed was very high at 4.9. Therefore, the candidates derived from the user procedures predictions were appropriate.

The necessity of the contents of the image shown was evaluated as 3.8. Many subjects found it useful, but some felt it was not needed. Since the need for such information varies with user skill, so the system must be adapt to levels of user skill.

6. Conclusions

We have proposed cooking support that recognizes user procedures and supports cooking procedures based on user progress using kitchen sensors. Our proposal consists of sensor information management, cooking procedure recognition, cooking procedure prediction, and information presentation. We implemented our proposal using a kitchen with ubiquitous sensors and confirmed that the system automatically provided appropriate information to users when predicted probability is high. The system shows candidate instructions for users to choose from when predicted probability is low.

Our proposal presented information to subjects using touch displays and speakers. Using mobile robots for physical support such as in transporting ingredients and utensils will be possible, and we are currently planning to use mobile robots as an intelligent environment medium.

References:

[1] M. Weiser, "The Computing for the Twenty-First Century," Scientific American, pp. 94-104, September, 1991.

[2] D.J. Moore, I.A. Essa, and M.H. Hayes III, "Exploiting Human Actions and Object Context for Recognition Tasks," Proc. of The 7th IEEE Int. Conf. on Computer Vision, pp. 80-86, 1999.

[3] T. Sato, Y. Nishida, and H. Mizoguchi, "Robotic Room: Symbiosis with human through behavior media," Proc. of Robotics and Autonomous Systems 18 Int. Workshop on Biorobotics, Human-Robot Symbiosis, ELSEVIER, pp. 185-194, 1996.

[4] M. Minoh, "Human Daily Life Support at a Ubiquitous Computing Home," Journal of Japanese Society for Artificial Intelligence, Vol.20, No.5, pp. 579-586, 2005.

[5] T. Fukuda, Y. Nakauchi, K. Noguchi, and T. Matsubara, "Sequential Human Behaviors Recognition for Cooking-Support Robots," Journal of Robotics and Mechatronics, Vol.17, No.6, pp. 717-724, 2005.

[6] Y. Yamakata, H. Ohara, A. Sawada, K. Kakusyo, and M. Minoh, "Peeling or Cutting Action and Its Target Food Product Recognition Based on Relationship between the Action and the Food Product," IEICE TRANSACTIONS on Information and Systems, Vol.90, No.9, pp. 2550-2561, 2007.

[7] Y. Yamakata, T. Shoji, K. Kakusyo, and M. Minoh, "Automatic Cooking Archiving with Spoken Dialogue with Assistant Agent," IEICE TRANSACTIONS on Information and Systems, Vol.90, No.10, pp. 2817-2829, 2007.

[8] I. Siio, R. Hamada, and N. Mima, "Kitchen of the Future and Its Applications," COMPUTER SOFTWARE, Vol.23, No.4, pp. 78-83, 2006.

[9] R. Hamada, et al., "Cooking navi: assistant for daily cooking in kitchen," Proc. of 13th Annual ACM Int. Conf. on Multimedia, pp. 371-374, 2005.

[10] Complete Cooking Recipes, "All about the Typical Recepes and Cooking Instructions in Home," NHK, 2004 (in Japanese).

[11] J.R. Quinlan, "Data Mining Tools See5 and C5.0," http://www.rulequest.com/see5-info.html

[12] A.R. Smith, "Color gamut transform pairs," Proc. of the 5th Annual Conf. on Computer Graphics and Interactive Techniques, pp. 12-19, 1978.

Name:
Sho Murakami

Affiliation:
Master Course Student, Department of Intelligent Interaction Technologies, Graduate School of Systems and Information Engineering, University of Tsukuba

Address:
Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:

2006- Master Course Student at University of Tsukuba

Main Works:

- Y. Nakauchi, T. Suzuki, A. Tokumasu, and S. Murakami, "Cooking Procedure Recognition and Support System by Intelligent Environments," Proc. of theIEEE Workshop on Robotic Intelligence in Informationally Structured Space (RiiSS 2009), pp. 99-106, 2009.
- S. Murakami, T. Fukuda, Y. Nakauchi, "Development of Cooking Procedure Recognition System by Using Ubiquitous Sensors and Machine Learning, Proc. of the JSME Workshop on Robotics And Mechatronics (Robomec'06), 2006.

Name:
Takuo Suzuki

Affiliation:
Master Course Student, Department of Intelligent Interaction Technologies, Graduate School of Systems and Information Engineering, University of Tsukuba

Address:
Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:

2007- Master Course Student at Toyohashi University of Technology
2008- Master Course Student at University of Tsukuba

Main Works:

- T. Okada, T. Imamura, T. Miyoshi, K. Terashima, Y. Yasuda, and T. Suzuki,"Muscle Strength Estimation using Musculo-skeletal Model for Upper-limb Rehabilitation," Journal of Robotics and Mechatronics, Vol.20, No.6, pp. 863-871, 2008.

Membership in Academic Societies:

- The Japan Society of Mechanical Engineers (JSME)
- The Society of Instrument Control Engineers (SICE)
- Japanese Society for Medical and Biological Engineering (JSMBE)
- The Institute of Electrical and Electronics Engineers (IEEE)

Name:
Akira Tokumasu

Affiliation:
Master Course Student, Department of Intelligent Interaction Technologies, Graduate School of Systems and Information Engineering, University of Tsukuba

Address:
Tsukuba, Ibaraki 305-8573, Japan

Main Works:
• "Development of Support System by Using Relations of Working Procedures," Proc. of 2008 JSME Conf. on Robotics and Mechatronics (ROBOMEC2008), 2P1-H02, 2008.

Membership in Academic Societies:
• The Robotics Society of Japan (RSJ)

Name:
Yasushi Nakauchi

Affiliation:
Associate Professor, Department of Intelligent Interaction Technologies, Graduate School of Systems and Information Engineering, University of Tsukuba

Address:
Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
1993- Research Associate at National Defense Academy
1994- Assistant Professor at National Defense Academy
1998-1999 Visiting Scientist at Carnegie Mellon University
1998-2003 Associate Professor at National Defense Academy
2003- Associate Professor at University of Tsukuba

Main Works:
• "A Social Robot that Stands in Line," Autonomous Robots, Vol.12, pp. 313-324, 2002.

Membership in Academic Societies:
• The Japan Society of Mechanical Engineers (JSME)
• The Robotics Society of Japan (RSJ)
• The Institute of Electrical and Electronics Engineers (IEEE)