A Cooking Recipe Recommendation System with Visual Recognition of Food Ingredients

http://dx.doi.org/10.3991/ijim.v8i2.3623

Keiji Yanai, Takuma Maruyama and Yoshiyuki Kawano
The University of Electro-Communications, Tokyo, Japan

***Abstract*—In this paper, we propose a cooking recipe recommendation system which runs on a consumer smartphone as an interactive mobile application. The proposed system employs real-time visual object recognition of food ingredients, and recommends cooking recipes related to the recognized food ingredients. Because of visual recognition, by only pointing a built-in camera on a smartphone to food ingredients, a user can get to know a related cooking recipes instantly. The objective of the proposed system is to assist people who cook to decide a cooking recipe at grocery stores or at a kitchen. In the current implementation, the system can recognize 30 kinds of food ingredient in 0.15 seconds, and it has achieved the 83.93% recognition rate within the top six candidates. By the user study, we confirmed the effectiveness of the proposed system.**

***Index Terms*—cooking recipe recommendation, food ingredients, object recognition, smartphone application**

I. INTRODUCTION

Recently, Web sites on cooking recipes such as cooks.com and BBC food search has become popular. Some of the people who cook use such sites to obtain information on cooking recipes. Because these sites are accessible from mobile phones as well as PCs, a user can access the cooking recipe sites at a grocery store as well as at home. However, to use these sites, a user has to input some keywords or select menu items to indicate his/her preferences on cooking menus. This may cause to prevent users from referring cooking recipe sites during shopping at grocery stores.

On the other hand, visual object recognition technology has been made much progress so far. Especially, generic object recognition, which is the technology that categories of the objects shown in a given image are recognized, have achieved tremendous progress. At the same time, open source libraries on object recognition such as the Open Computer Vision library (OpenCV) has spread widely. With such libraries, we can effectively implement an object recognition system not only on PCs but also on mobile devices such as iPhones and Android smartphones. In addition, computational power of mobile devices has progressed greatly, which is equivalent to one of a-few-years-ago PCs. A multiple-core CPU is common as a smartphone CPU. Due to recent progress of object recognition technology as well as recent progress of computational power of mobile devices, visual object recognition on mobile devices in a real-time way becomes possible.

Based on these situations, in this paper, we propose a cooking recipe recommendation system on a mobile device employing object recognition for food ingredients such as vegetables and meats. The proposed system carries out object recognition on food ingredients in a real-time way on Android-based smartphones, and recommends cooking recipes related to the recognized food ingredients. By pointing a mobile phone camera toward food ingredients, a user can receive a recommendation recipe list instantly. We designed and implemented the system to be used easily and intuitively during shopping at grocery stores or supermarkets as well as before cooking at home. Figure 1 shows an example usage of the proposed system that a user is pointing a mobile phone camera to tomatoes at a grocery store and searching for the cooking recipes related to tomatoes.

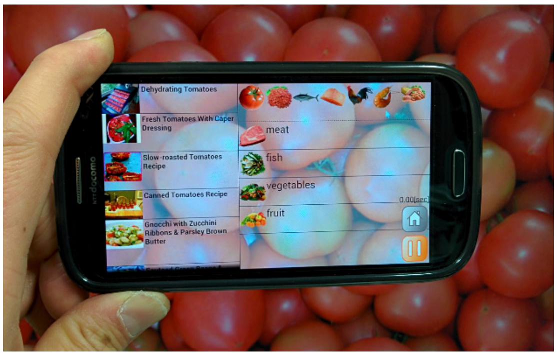

Dehydrating Tomatoes
Fresh Tomatoes With Caper Dressing
Slow-roasted Tomatoes Recipe
Canned Tomatoes Recipe
Gnocchi with Zucchini Ribbons & Parsley Brown Butter
meat
fish
vegetables
fruit
0.00[sec]
NTT docomo

Figure 1. An image on the proposed system. A user points a mobile phone camera to food ingredients at a grocery stores, and then the system advises cooking recipes based on the recognized ingredients instantly.

To speed up object recognition for enabling the system to recommend cooking recipes in a real-time way, the system uses color-histogram-based bag-of-features extracted from multiple frames as an image representation and a linear kernel SVM as a classifier. We built 30 kinds of food ingredient short video database for the experiments. With this database, we achieved the 83.93% recognition rate within the top six candidates. In the experiment, we made user study by comparing mobile recipe recommendation systems with/without visual recognition of food ingredient.

In the rest of this paper, we describe related work in Section II. In Section III, we explain the overview and detail of the proposed system. The method to recognize food ingredients used in the proposed system is described in Section IV. Section V shows experimental results and user study. We conclude this paper in Section VI.

II. RELATED WORK

In this section, we introduce related works in terms of mobile object recognition, image recognition for food ingredients as well as cooking recipe recommendation.

As commercial services on image recognition for mobile devices, Google Goggles[1] is widely well-known. Google Goggles work as an application on both Android and iPhone, which can recognize letters, logos, famous art, cover pages of books and famous landmarks in photos taken by users with object recognition technology. Since it is mainly based on specific object recognition method employing local feature matching, it is good at rigid objects such as landmarks and logos. However, it cannot recognize generic objects such as animals, plants and foods at all.

As a similar work, Lee et al. [6] proposed a mobile object recognition system on a smartphone, which recognized registered objects in a real-time way. They devised descriptors of local features and their matching method for real-time object recognition. On the other hand, Yu et al. [10] proposed a mobile location recognition system which recognizes a current location by local-feature-based matching with street-view images stored in a database. In their work, they proposed automated Active Query Sensing (AQS) method to automatically determine the best view for visual sensing to take an additional visual query. All of these systems aimed local-feature-based specific object matching, while we focus on generic object recognition on food ingredients.

Next, we explain some works on image recognition on food ingredients. In Smart Kitchen Project [5] leaded by Minoh Lab, Kyoto University, which aims to realize a cooking-assisted kitchen system, image recognition for food ingredients is used. This project includes image classification and tracking on food ingredients during cooking. While in this project food ingredient recognition is used for cooking assistance, in our work it is used for cooking recipe search.

Regarding works on cooking recipe recommendation, text-based methods have been studied so far. Ueda et al. proposed a method to recommend cooking recipe based on user's preference [9], and Shidochi et al. worked on finding replaceable ingredients in cooking recipe [8]. Akazawa et al.[1] proposed a method to search cooking recipes based on food ingredients left in a refrigerator. The recommended recipes are ranked in the order considering consumption date and remaining amount of ingredients. In the current work of ours, we did not use the detail information on available ingredients. We plan to take into account the conditions of ingredients on amounts, nutrition and prices for future work.

III. PROPOSED SYSTEM

In this section, we explain an overview and detail of the proposed system.

A. *Overview*

The objective of this work is to propose a mobile system which assists a user to decide what and how to cook using generic object recognition technology. We assume that the proposed system works on a smartphone which has built-in cameras and Internet connection such as Android smartphones and iPhones. We intend a user to use our system easily and intuitively during shopping at grocery stores or supermarkets as well as before cooking at home. By pointing food ingredients with a mobile phone built-in camera, a user can receive a recipe list which the system obtained from online cooking recipe databases instantly. With our system, a user can get to know the cooking recipes related to various kinds of food ingredients unexpectedly found in a grocery store including unfamiliar ones and bargain ones on the spot.

To do that, the system recognizes food ingredients in the photos taken by built-in cameras, and search online cooking recipe databases for the recipes which need the recognized food ingredients.

As an object recognition method, we adopt bag-of-features with SURF and color histogram extracted from not single but multiple images as image features and a linear kernel SVM with the one-vs-rest strategy as a classifier.

B. *Processing Steps*

As mentioned before, our system aims to search for cooking recipes during shopping at grocery stores. In this subsection, we describe the flow of how to use the proposed system from taking photos of food ingredients until watching the recipe pages a user selected. Figure 2 shows the flow.

[1] Point a smartphone camera toward food ingredients at a grocery store or at a kitchen. The system is continuously acquiring frame images from the camera device in the background.

[2] Recognize food ingredients in the acquired frame images continuously. The top six candidates are shown on the top-right side of the screen of the mobile device. (See Figure 3.)

[3] Search online cooking recipe databases with the name of the recognized food ingredient as a search keyword, and retrieve a menu list. If a user like to search for recipes related to other candidates than the top one, the user can select one of the top six ingredients by touching the screen.

[4] Display the obtained menu list on the left side.

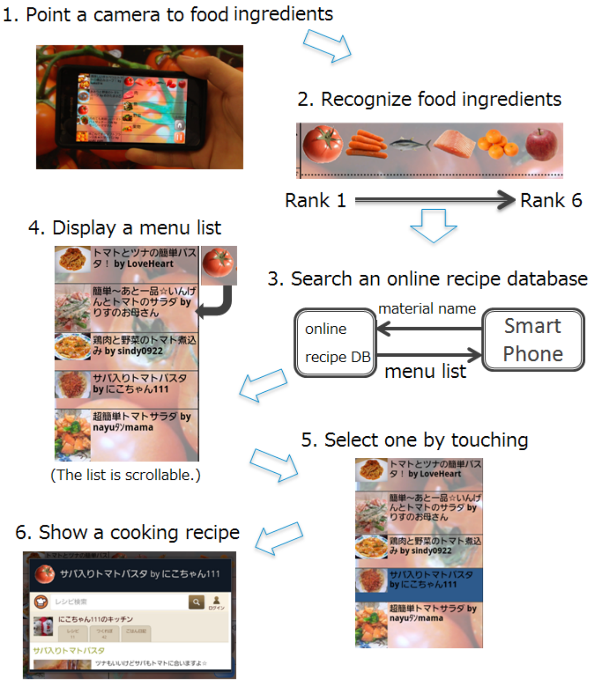

1. Point a camera to food ingredients
2. Recognize food ingredients
Rank 1
Rank 6
3. Search an online recipe database
online
recipe DB
material name
menu list
Smart
Phone
4. Display a menu list
トマトとツナの簡単パスタ！ by LoveHeart
簡単～あと一品☆いんげんとトマトのサラダ by りすのお母さん
鶏肉と野菜のトマト煮込み by sindy0922
サバ入りトマトパスタ by にこちゃん111
超簡単トマトサラダ by nayuタン/mama
(The list is scrollable.)
5. Select one by touching
6. Show a cooking recipe
サバ入りトマトパスタ by にこちゃん111
レシピ検索
にこちゃん111のキッチン
サバ入りトマトパスタ

Figure 2. Processing flow of the proposed system.

[1] http://www.google.com/mobile/goggles/

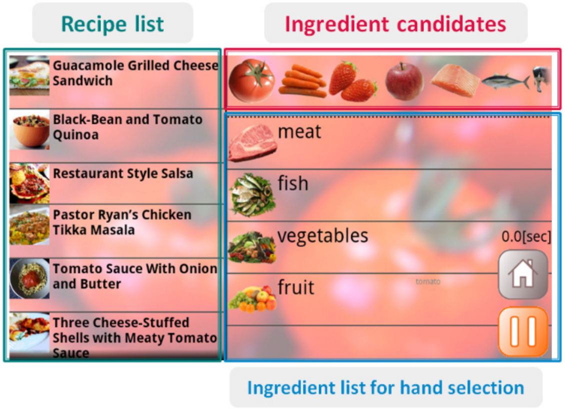

Recipe list
Ingredient candidates
Guacamole Grilled Cheese Sandwich
Black-Bean and Tomato Quinoa
Restaurant Style Salsa
Pastor Ryan's Chicken Tikka Masala
Tomato Sauce With Onion and Butter
Three Cheese-Stuffed Shells with Meaty Tomato Sauce
meat
fish
vegetables
fruit
0.0[sec]
Ingredient list for hand selection

Figure 3. The system screen. On the top right of the screen, recognized results are shown as food ingredient candidates. On the left side, the recommended recipe list is shown. A user can select recipes to see the details by touching the screen.

[5] Select one menu from the menu list. A user can see other menus than ones shown on the screen initially by scrolling.

[6] For the selected menu, display the corresponding cooking recipe including a list on necessary ingredients and seasonings and a cooking procedure on the pop-up window. Basically, the recipe page in the original recipe site will be shown.

Typically, a user uses the proposed system according to the above steps from one to six. **Error! Reference source not found.**shows the system screen.

C. *Search Online Cooking Recipe Databases*

Instead of preparing our own cooking recipe database, we use Web APIs of commercial cooking recipe sites on the Web such as CookPad[2] and PunchFork[3]. CookPad is a Japanese food recipe site where all the information is written in Japanese language, while PunchFork mainly focuses on Western food recipes which is operated by a US company.

Currently, we send the names of recognized food ingredients as search terms as they are, and obtain research results on cooking recipes in which the recognized food ingredients are needed to cook. Re-ranking of the returned results from the Web API of cooking recipe sites considering various elements including prices, amounts and user's preferences is our future work.

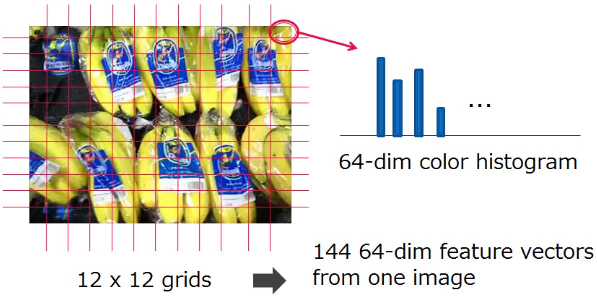


Figure 4. Grid-based extraction of local color histograms.

IV. IMAGE RECOGNITION METHOD

In this section, we explain a method on image-based food ingredient recognition, which is a core part of the proposed system, regarding image features, image representation, and image classifier.

A. *Image Features*

For real-time object recognition on a mobile device, choice of image features is important in terms of accuracy and speed, both of which are trade-off in general. Recently, new local features which are suitable for a mobile device such as BRIEF[3] and ORB[7] are proposed. They require small memory and run very fast because of binary-based descriptors. However, all the new features for a mobile device intends instance-level specific object recognition based on local feature matching. In our system, we need to carry out category-level generic object recognition. To prevent information loss due to binarization of feature vectors, we use SURF local feature.

SURF is an invariant local feature for scale, rotation and illumination change. When extracting SURF features, we use dense sampling where all the local features are extracted from multi-scale grids as well as a fast Hessian detector which is a default key-point detector of SURF.

In addition, for recognition of food ingredients, color is regarded as important information as well. We also extract grid-based color histograms. As shown in Figure 4, we divide an image into 12×12 grids, and extract a 64-bin color histogram from each block with dividing the space into 4×4 bins. Totally, we extract 144 64-dim color feature vectors from one image. We regard each color feature vector as a local color feature, and convert them into bag-of-feature representation in the same way as SURF features in the next step. Note that we used three kinds of color spaces including RGB, HSV and La*b* in the experiments.

B. *Bag-of-Features Representation*

Bag-of-Features (BoF) is a standard feature representation to convert a set of local features into one feature vector, since it has excellent ability in the context of category-level generic object recognition in spite of its simplicity.

To convert a set of local feature vectors into a BoF vector, we vector-quantize them against the pre-specified codebook. After that, all the BoF vectors are L1-normalized. In the experiments, we built a 1000-dim codebook by k-means clustering with local features sampled from training data offline on a PC.

In this work, we can use multiple frames to build a BoF vector, since we acquire frame images from the built-in camera continuously. Therefore, in the experiments, we aggregated local features extracted from five frames at most, and convert them into one BoF vector.

C. *Image Classifier*

In this paper, we use a Support Vector Machine (SVM) which is the most common classifier. It is common to use non-linear kernels such as a RBF-χ^2 kernel with a SVM in category-level object recognition task, because of its high classification performance. However, a non-linear kernel SVM is computationally expensive compared to a linear kernel SVM. In case of classification step, the computation cost of a non-linear kernel SVM is *O(dn)*, while that of a linear kernel SVM is *O(d)* where *d* and *n* represents

[2] http://cookpad.com/
[3] http://punchfork.com/

the dimension of feature vectors and the number of support vectors which is typically proportional to the number of training samples, respectively. Since we prioritize low computational cost for real-time recognition, we adopt a liner kernel SVM.

A liner kernel $K(x,y)$ is represented in the following function, which is equivalent to an inner product of two vectors.

$$K(x,y) = x \cdot y \tag{1}$$

When $x, y(x), N, x_i, w_i$ and b represents an input feature vector, the output of a SVM classifier, the number of support vectors, a support vector, the weight of the corresponding support vector, and a bias scalar value, respectively, the equation of a linear SVM classifier can be transformed as follows:

$$y(x) = \sum_{i=1}^{N} w_i K(x, x_i) + b \tag{2}$$

$$= \sum_{i=1}^{N} w_i x \cdot x_i + b$$

$$= \mathrm{x} \cdot \sum_{i=1}^{N} w_i x_i + b$$

$$= \mathrm{x} \cdot \mathrm{v} + \mathrm{b} \tag{3}$$

where $\mathrm{v} = \sum_{\mathrm{i}=1}^{\mathrm{N}} w_i x_i$. As shown above, we can evaluate one classifier with only the computation of one inner product between two vectors and addition of a bias scalar value. In the experiments, to recognize 30 kinds of food ingredients, we adopt the one-vs-rest strategy.

V. EXPERIMENTS

A. Data Collection and Experimental Setting

As a data set for the experiments, we collected 10 short videos per ingredient category for 30 kinds of food ingredients at grocery stores in Tokyo, which are listed in TABLE I. Since we use multiple frame object recognition, we collected short videos instead of still images. Each of the videos was recorded for about 5 seconds in 25 fps with the VGA (640x480) resolution. In the experiments we carried out evaluation of object classification performance with 10-fold cross validation.

In the experiments, we set the parameters as shown in TABLE II.

B. An Example Usage of the Proposed System

Before showing the evaluation results, we show a typical usage of the proposed system at a grocery store in Figure 5. In these cases, we used the implemented system on Samsung Galaxy S2 (1.5GHz dual core, Android 2.2). On this device, it took 0.15 seconds to recognize an ingredient with the built-in camera in case of using only color features.

TABLE I.
30 KINDS OF FOOD INGREDIENTS IN THE DATA SET.

Types	ingredients
Fish (5)	tuna, squid, octopus, shrimp, salmon
Meat (5)	beef, pork, chicken, minced meat, sausage, ham
Vegetable (13)	mushroom, potato, eggplant, carrot, radish, tomato, cucumber, cabbage, green onion, onion, Chinese cabbage, lettuce, Shiitake mushroom
Fruit (6)	apple, strawberry, pineapple, orange, banana, grapefruit

TABLE II.
PARAMETER SETTINGS IN THE EXPERIMENTS.

Parameter setting in the experiments

Multi-scale-grid-based SURF extraction
4 scales ($12 \times 12, 24 \times 24, 48 \times 48, 96 \times 96 pixels$)

Grid-based color histogram
1 scales (dividing a image into $12 \times 12 grids$)

Color space of color histogram
RGB, HSV, La*b*

multi-frame feature extraction
$1, 2, 3, 4, 5$ frames

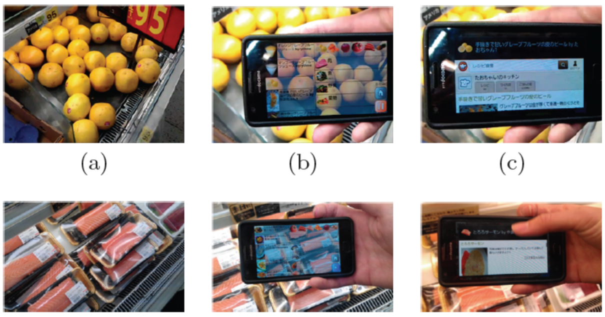

(a) (b) (c)
(d) (e) (f)

Figure 5. Typical situations with the proposed system: (a) The target is a "grapefruit'". (b) A "grapefruit" is successfully recognized as the top candidate, and the cooking menus related to it are shown on the screen. (c) The selected recipe related to "grapefruit" is shown. (d) The target is a "salmon". (e) A "salmon" is ranked in the third. Then, "salmon" is selected by touching the screen. (f) The selected recipe related to "salmon'" is shown.

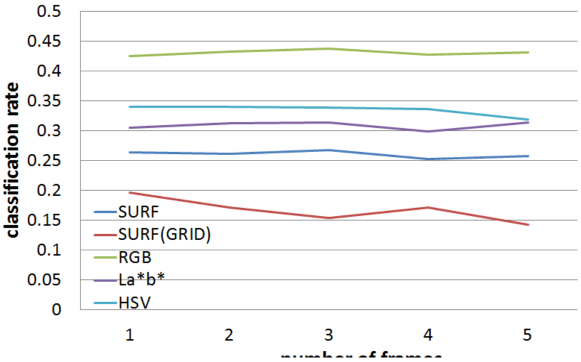


Figure 6. Classification rate using *n* frames

C. Evaluation of Object Classification Performance

We evaluated the classification accuracy with various settings in terms of image feature extraction.

At first, we made experiments with single image features, which are SURF with a fast Hessian detector (default detector), SURF with multi-scale grids, RGB, HSV and La*b*, varying the number of frames to build a BoF vector. We show the results in Figure 6, which indicates that the difference depending on the number of frames is limited, and the best classification rate, 43.78%, is achieved in case of a RGB color feature with three frames.

Figure 7 shows the results of single features with a single frame, a RGB color feature with three frames, and the combination of SURF and RGB with three frames in a bar graph. Although the combination of SURF and RGB with three frames achieved the best result, 44.92%, the difference to the result by only RGB feature is only 1.14%. The reason that SURF feature does not work as well as RGB color is that the dataset contains many blurred or reflected frames as shown in Figure 10, from which it is difficult to extract gradient-based features such as SURF effectively.

Figure 8 shows the classification rate within the top *k* candidates in case of RGB, SURF and combination of RGB and SURF with three frames. This shows that the result by RGB and the result by combination of RGB and SURF are almost equivalent. Therefore, in the release version of the application of the proposed system, we use only RGB color features. Because the top six candidates can be shown on the screen at the same time in the current implementation, the classification rate within six is important, which is 83.93%.

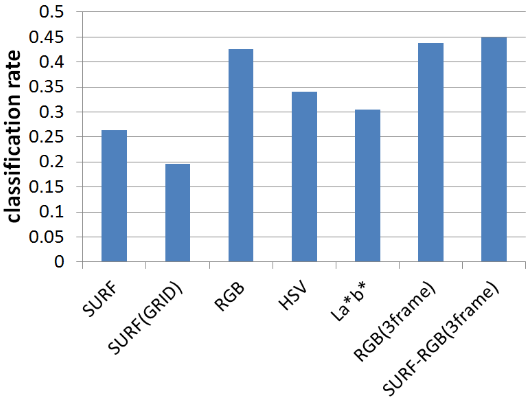


Figure 7. Classification rate by each of the image features.

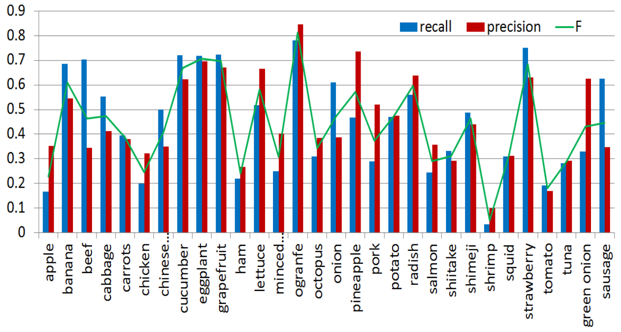


Figure 8. Classification rate of the top *k* candidates

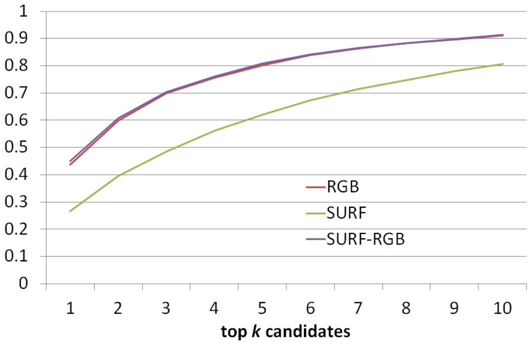


Figure 9. Precision, recall rate and F-measure for each of thirty kinds of the food ingredients.

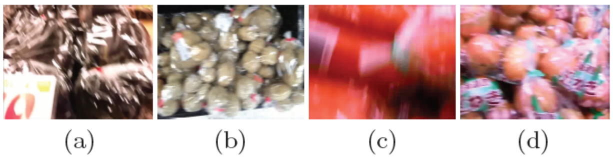


Figure 10. Example photos in which recognition failed due to reflectance and blurring.

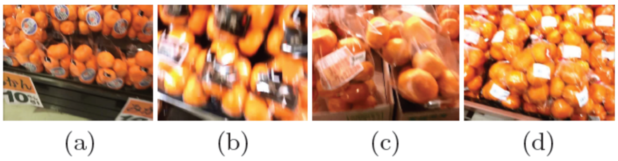


Figure 11. Easy food ingredients to recognize: "orange."

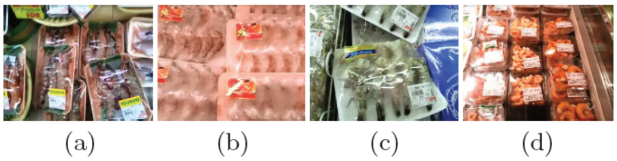


Figure 12. Difficult food ingredients to recognize: "shrimp."

Figure 9 shows the precision, the recall rates and the F-measure for each of the 30 kinds of ingredients by combination of RGB color and SURF with three frames, which achieved the best result on the average. "Orange" achieved the best result (Figure 11. On the other hand, "shrimp" achieved the worst (Figure 12. Their appearance depends on how to pack greatly. This is completely different from "shrimp" images in the common image database such as the ImageNet database[4].

D. User Study

In this subsection, we show the results of user study employing five subjects.

At first, we recorded the times to search for the recipes related to the given real food ingredients with the proposed image-based method as well as by selecting ingredients by hand. Next, we asked them three questions on how easy to use the system, how accurate ingredient recognition was, and which is easier to use, image recognition or selecting by hand. We collected all the answers in the five-step evaluation. For this study, we prepared three kinds of real food ingredients.

[4] http://image-net.org/

Figure 13 shows the times to obtain the cooking recipes related to the given food ingredients both in case of using object recognition and in case of selecting ingredients by touching the screen. The median of the times are 7.3 seconds by hand and 8.5 seconds by image recognition, respectively. This is because six cases by image recognition took more than fifteen seconds. However, the cases which took only less than two seconds were twice by image recognition, but none by hand. This shows that if image recognition works successfully, image recognition is faster and more effective than hand selection. We think this tendency gets more remarkable, if the number of food ingredients to be recognized becomes larger.

To select an ingredient from a 30-kind list by touching the screen, a user has to select hierarchical menus and sometimes has to scroll the menus to find out the given ingredient. On the other hand, a user sometimes has to continue to change the build-in camera position and direction until the correctly-recognized ingredient appears within the top six candidates on the screen, although the rate within six candidates was 83.93%. For these reasons, both image-based method and hand-based methods sometimes took more than ten seconds.

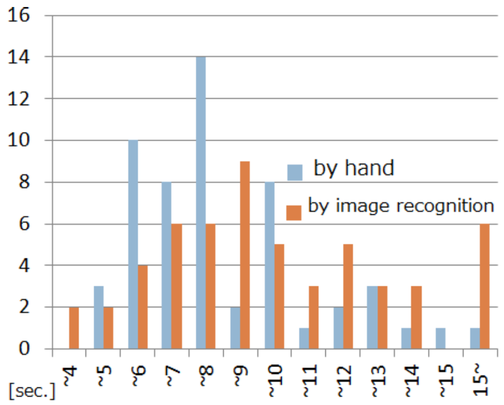
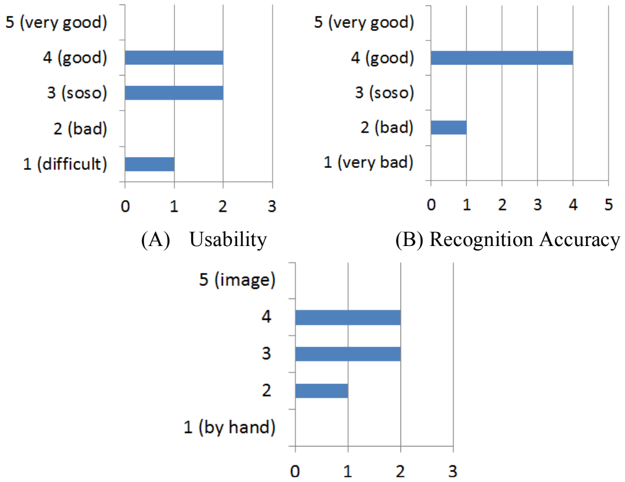


Figure 13. A graph showing the distribution on time (x-axis, seconds) vs. frequency (y-axis) to select a recipe by hand and by image recognition.



(A) Usability

(B) Recognition Accuracy

(C) Which is better, image recognition or by hand?

Figure 14. The results of 5-step questions on usability, accuracy and preference between the proposed recognition-based system and the manual baseline system.

Next, we show the five-step evaluation results of the questions on usability, accuracy of image recognition, and comparison of both methods in Figure 14(A), Figure 14(B) and Figure 14(C), respectively.

Regarding usability, two subjects answered the proposed system was better, while one subject answered the manual system was much better. This is because the latter subject experienced that it took more than 15 seconds to reach the correct food ingredient names several times. In fact, the user interface of the current system is not so sophisticated that an inexperienced user revises incorrectly-recognized results easily when a correct ingredient name is not shown within the top six candidates on the screen. In fact, in this work, we gave priority to the image recognition part rather than the user interface. Improvement of the user interface is part of our future work.

Regarding the subjective feeling on recognition accuracy, four subjects answered it was better than they expected. Because the rate that correct names of ingredients are shown in the candidate lists is more than 80%, they were satisfied with recognition accuracy.

Regarding the last question on preference between two systems, two subjects answered the proposed recognition-based system was better, while one subjects answered the manual baseline system was better. Although four subjects were satisfied with recognition accuracy, only two voted on the proposed system. Other factors than recognition such as user interface seem to affect users' preference.

Overall, the evaluation results on the recognition-based system is slightly better than the evaluation results on the hand-based baseline system, although the difference is not so large. We obtained some positive comments from the subjects. "It is convenient to be able to search for cooking recipes during shopping, when bargaining ingredients are found unexpectedly." "If the accuracy of recognition is improved and the number of kinds of the ingredients is increased, the system will be much more practical."

VI. CONCLUSIONS

In this paper, we proposed a mobile cooking recipe recommendation system with food ingredient recognition on a mobile device, which enables us to search for cooking recipes only by pointing a built-in camera to food ingredients instantly. To our best knowledge, this is the first work which integrates visual object recognition into a mobile cooking recipe recommendation system. Regarding recognition performance, for 30 kinds of food ingredients, the proposed system has achieved the 83.93% classification rate within the top six candidates. From the user study, it is turned out that the system was effective in case that food ingredient recognition works well.

For future work, we plan to improve the system in terms of object recognition, recipe recommendation and the system user interface. Regarding object recognition, we would like to achieve 90% classification rate within the top six candidates for the 100 category food ingredients by adding other image features and segmenting food ingredient regions from background regions. Regarding recipe recommendation, we plan to implement recipe search considering combination of multiple food ingredients, nutrition and budgets.

Note that the application for Android smartphones of the proposed system can be downloaded from http://mirurecipe.mobi/e/ .

REFERENCES

[1] Y. Akazawa and K. Miyamori. Cooking recipe search system considering food materials remained in a refrigerator. In Proc. the 3rd Forum on Data Engineering and Information Management, 2011 (in Japanese).

[2] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. In Proc. of European Conference on Computer Vision, pages 404-415, 2006.

[3] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, BRIEF: Binary robust independent elementary features. In Proc. of European Conference on Computer Vision, 2010.

[4] G. Csurka, C. Bray, C. Dance, and L. Fan. Visual categorization with bags of keypoints. ECCV WS on Statistical Learning in Computer Vision, pages 1-22, 2004.

[5] A. Hashimoto, N. Mori, T. Funatomi, Y. Yamakata, K. Kakusho, and M. Michihiko. Smart kitchen: A user centric cooking support system. In Proc. of International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, pages 848-854, 2008.

[6] T. Lee and S. Soatto. Learning and matching multiscale template descriptors for real-time detection, localization and tracking. In Proc. of Computer Vision and Pattern Recognition, pages 1457-1464, 2011.

[7] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An efficient alternative to SIFT or SURF. In Proc. of International Conference on Computer Vision, 2011.

[8] Y. Shidochi, T. Takahashi, I. Ide, and H. Murase. Finding replaceable materials in cooking recipe texts considering characteristic cooking actions. In Proc. of ACM MM WS on Multimedia for Cooking and Eating Activities (CEA'09), pages 9-14, 2009.

[9] M. Ueda, M. Takahata, and S. Nakajima. User's food preference extraction for cooking recipe recommendation. In Proc. of the 2nd Workshop on Semantic Personalized Information Management: Retrieval and Recommendation, 2011.

[10] X. F. Yu, R. Ji, and S.-F. Chang. Active query sensing for mobile location search. In Proc. of ACM Multimedia, pages 3-12, 2011.

AUTHORS

K. Yanai received B.Eng., M.Eng. and D.Eng. degrees from the University of Tokyo in 1995, 1997 and 2003, respectively. From 1997 to 2006, he was a research associate at the University of Electro-Communications, Tokyo. Currently, he is an associate professor at the University of Electro-Communications, Tokyo. (e-mail: yanai @ cs.uec.ac.jp).

T. Maruyama received B.Eng. and M.Eng. from the University of Electro-Communications, Tokyo in 2010 and 2012, respectively. Currently, he is working for a company developing mobile applications. (e-mail: maruya-t @ mm.cs.uec.ac.jp).

Y. Kawano received B.Eng. from the University of Electro-Communications, Tokyo in 2013. Currently, he is a master-course student at the graduate school of the University of Electro-Communications, Tokyo. (e-mail: kawano-y @ mm.inf.uec.ac.jp).

This work was supported in part by JSPS KAKENHI Grant Number 23650044, 24300038, 25135714. Manuscript received 28 February 2014. Published as re-submitted by the authors 05 April 2014.