KUSK OBJECT DATASET: RECORDING ACCESS TO OBJECTS IN FOOD PREPARATION

Atsushi Hashimoto, Masaaki Iiyama, Shinsuke Mori, Michihiko Minoh

Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto, Japan

ABSTRACT

This study aims to construct a KUSK Dataset's extension that provides records of chef's touching and releasing action to objects, which we call "*access to objects*," in his/her food preparation. The records of access to object are known as a key evidence for understanding chef's activity in food preparation. In the dataset, we provide object images as well as the records of access to object. The data are obtained by manual annotation and by automatic processing. As a result of annotation, we collected 4391 object images from 57 cooking observations. We also confirmed that the CNN-based state-of-the-art method reached 74.15% accuracy on average in recognizing objects on a cooking counter.

***Index Terms*—** Cooking and Eating Activities, Dataset, Object Recognition, Annotation

1. INTRODUCTION

It is the main goal in computer science to make computers understand every event observed by sensors and generate explanatory texts. Encouraged by a large-sized corpus obtained from the Web and an accurate corpus custom-built by crowd sourcing, image-text translation has been actively studied in recent years [1, 2, 3]. There are also a few methods that translate a short video clip into texts [4]; however, as long as the authors know, no methods exist as yet to translate a video with multiple events into text. In order to obtain an accurate translator for such long video, it is a necessary process to prepare an amount of pairs of video and text that are aligned in each event, namely the video should be segmented into clips and each clip has corresponding textual descriptions for an event. There has not been so much research as yet in the area of video-text alignment [5].

For the study of video-text alignment and translation, food preparation is a practical and applicative subject. Food preparation is a sequence of tasks in which materials are processed, divided, and merged to finish a product. This is the general framework in manufacturing activities. Similarly, recipes follow a general form of procedural text in the sense that they are useful to describe any type of food preparation. Thus, a method that matches a recipe text and tasks in food preparation activity can be applied to other manufacturing processes as well.

As a key evidence to understand events in food preparation, we focus on humans touching and releasing objects, which we call "*access to objects*" [6]. Our previous work [6] achieved an accuracy of approximately 70% when forecasting the next sequence of action from access to object with the help of recipe information. In this study, we aim to construct KUSK Object Dataset: a KUSK Dataset's extension that containing the records of access to objects, for encouraging studies of the video-text alignment problem in food preparation.

2. RELATED DATASETS

The past success of image-text translation was led by PASCAL sentence dataset [7], which provided both annotation text for images and image processing results for the same images. Because a collaboration of computer vision (CV) and natural language processing (NLP) is necessary for this task, the provided CV results were shared by several NLP researchers [1, 2]; this boosted the studies of image-text translation in the early stage. Modeling after the success in Pascal-sentence dataset, we also provide a dataset that is supposed to be used by NLP researchers, as well as provide a baseline in detecting access to object for CV researchers.

There are two main components that describe a food preparation activity: the chef's physical motions, and the status of objects and equipment on the cooking counter. TUM Kitchen Data Set [8] and CMU Multi-Modal Activity Database [9] focused on motions that are recorded by cameras placed on the ceiling [8], first person vision cameras [9], and motion captures [8, 9].

Some datasets have recorded both motions and the status of the cooking counter by RGB-D cameras [10, 11]. An ideal way is to observe food preparation activities by placing cameras behind the cooking counter; however this setting can only be done with an island kitchen. The Breakfast Actions Dataset [12] is a dataset that attempts to observe chef's motions and cooking counter from a camera attached on the side wall. Although this setting will face more self-occlusions (in observing motions) and inter-occlusions (in observing cooking counter), it is available with I-shaped and L-shaped kitchens.

There are datasets that place more value on observing cooking counter rather than the chef's motion. 50 Salads dataset [13] observes food preparation activities by cameras set on the ceiling. To further enrich information on the chef's motion, wearable 3D accelerometers are used together with

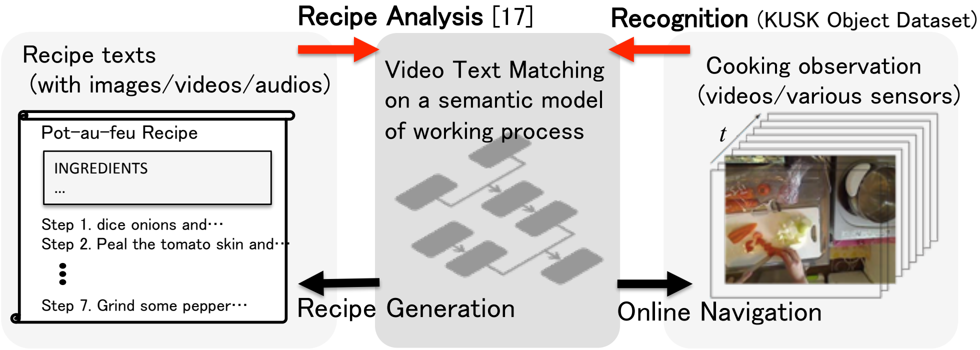

Fig. 1. Usecases of our dataset.

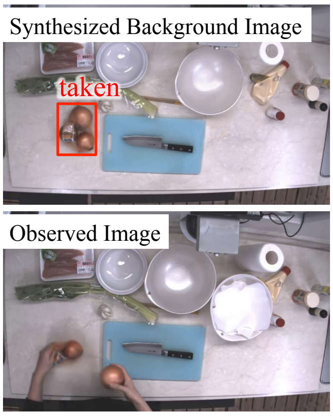

Fig. 2. Annotation Example.

cameras. Similarly, KUSK Dataset [14] [1] supplementarily provides videos from a camera on the side wall and various sensors, such as flow meters for the water supply and sewerage systems and power meter for the induction heaters in addition to cameras on the ceiling. In addition to those sensors, KUSK dataset uses load sensors to observe mechanistic interactions between the chef and the cooking board, which is unobservable by any cameras.

We prepare our dataset as an extension of KUSK Dataset, namely, KUSK Object Dataset, because KUSK Dataset collaborates with Flow Graph Corpus [15][2]. Flow Graph Corpus is a publicly available dataset providing workflow structure of recipe texts obtained by a NLP technique. The workflow structure of any observations in KUSK Dataset's 20 recipes is available in Flow Graph Corpus. Providing the record of access to object in videos in KUSK Dataset yields many options that can be used in a variety of studies: as an input to match recipe text and processes of food preparation, as an input to analyze recipe texts with observations, and as a baseline of object recognition in food preparation (Fig 1).

3. SEMI-AUTOMATIC ANNOTATION FOR ACCESS TO OBJECTS

A task of video annotation is generally time consuming. In our case, the chef's access to objects was nearly 100 times or more during the 30 minutes of food preparation [16]. Although the frequency of such accesses depends on each recipe, it is too costly to annotate every access to objects manually.

To reduce the burden of the annotation task, we hire an object detection method proposed in [16]. This method is based on background subtraction, which does not require any training data of objects. Because we do not have enough annotated images for food preparation, it is necessary to work without training data. Furthermore, this method detects objects with the label of "put" and "taken," which are respectively corresponding to "release" and "touch."

Using the results from this method, we semi-automate the annotation task. The task comprises three manual steps: correcting errors in the output from object detection, listing up vocabulary of object classes for each recipe, and assigning the object class to each object, which is automatically detected and manually confirmed at the first step. These three steps do not require annotators to scan videos thoroughly.

We will now describe the three manual steps in more detail. In the first step, the results by [16] might include simple missed/false-detection, failures in labeling "put" and "taken," and errors such that two or more objects are detected as an object or an object as several objects. To correct these errors, we instructed annotators to compare two images: the observed image at each detection and the background image that includes only objects on the tabletop and no body parts of chef and no objects in chef's hands (Fig. 2). In other words, the background image represents the state of objects on the tabletop before the access, and annotators can check whether the object is still on the tabletop or in hand by comparing these two images. Regardless of failures in the detected region, annotators are requested to draw a rectangle that covers every newly-put object and newly-taken object, with a label of "put" or "taken."

In the second step, annotators simulate in his/her head how to cook the dish and list up object classes that might appear in cooking the recipe. In this step, they are allowed to use any type of cooking utensils but are not allowed to rearrange the recipe by replacing, adding, or omitting ingredients and seasonings. For normalizing class names in the vocabulary, we limit the names of cooking utensils, seasonings and ingredients to those listed in the cooking ontology provided by Nanba et al. [17].

There were some ingredients, seasonings, and utensils that are not listed in the ontology. We instructed annotators to notify the authors whenever an annotator felt it necessary to annotate a new class name. We add the name as a new class when the class name is not overlapped by any other objects in

[1] `http://kusk.mm.media.kyoto-u.ac.jp/`

[2] `http://plata.ar.media.kyoto-u.ac.jp/data/recipe/home.html`

the ontology.

There were other types of exceptions. The first type is background objects, which is hardly related to food preparation, or difficult to distinguish even for a human. Dish detergents and sponges do not appear in recipes. Stem ends of foods, any other parts that we do not eat, or caps of seasoning bottles are sometimes hard to distinguish. Such objects are treated as a background class.

The other type of exception is a mixture of ingredients. Those ingredients have no explicit names, thereby they have no classes. To identify difference in such mixtures, annotator will need to check what was done to it previously in the video. To avoid such video check, we labeled such objects as "mixture of ingredients" while keeping its recipe ID (e.g., a mixture of ingredients appeared in recipe 1 has a label of "mixture-of-ingredients/recipe-1"). Although this way of annotation cannot distinguish the different types of mixture in a recipe, recipe IDs make it possible to distinguish mixtures that have appeared in different recipes at least. To automate the annotation on the mixture of ingredients in a recipe, tracking the ingredients will be a necessary requirement. This is reserved for our future work.

In the third step, annotators assign an appropriate object class for each rectangle given in the first step. Here, the vocabulary of object classes is given by the second step. All the above steps are checked by different annotators for quality control. Namely, when two annotators annotated differently to the same target, the annotated data was passbacked to another annotators. When the same target was passbacked repeatedly, a supervisor, which is one of the authors, intervene to the annotation task.

Fig. 3 shows examples of object images cropped on the basis of annotated rectangles. From these samples, we found that there are several difficulties in recognizing objects observed in food preparation. Firstly, ingredients appear in several different states. Eggs are typical examples. Secondly, containers, such as the chopping board and the pan, would be observed with their contents in many of the observations. Although containers do not appear in a recipe, the type of used containers implicitly indicates the process that a chef is performing. In this sense, containers are not a trivial recognition target. Most of the pre-existent methods for object detection do not assume that detection targets are laid under other salient objects. Containers are also an interesting target of recognition in food preparation.

4. OBJECT RECOGNITION AS A FULL AUTOMATIC PROCCESSING RESULT

Although annotation data are useful for training classifiers, it is not an output from an automatic computer processing. To provide data that is obtained automatically, we applied the state-of-the-art object classifier [18] both to the images obtained as a result of annotation and to the images detected by [16].

4.1. Settings for object recognition

Recently, many different structures of a convolutional neural network (CNN) have been proposed, and pre-trained models for many of those structures are available. In this study, we use the structure of Res-Nets [18] with 152 layers, which achieved the best score in ILSVRC and COCO 2015 competitions. These pre-trained models, however, are trained on a general image dataset and are not specified to food preparation.

There are two different approaches to fit the model to a specific problem: fine-tuning the original model and the use of a custom-trained linear support vector machine (SVM) instead of the last full-connection layer. We adopted the latter approach because the vocabulary of recognition targets differs for each recipe. To optimize the vocabulary for each recipe, we should prepare classifiers of the same number with recipes. Because there can be a large number of recipes in a system, it does not seem to be practical to store parameters for the number of CNNs. Thus, we assume a system with custom-trained SVMs, which costs less disk and memory resources than CNNs.

The pre-trained model requires input images to be 224 × 224 pixels. Resizing small object images might cause a bad effect to the feature extraction. Hence, we set the minimum cropping size of objects to be 128 × 128 pixels in order to avoid enlarging object images to more than twice their size while keeping the cropped image to contain only the target object without its neighbors.

We used only object images with a "put" label as training/test data. Object images with "taken" label must have the corresponding sample, which is the same object in the same pose with a "put" label because of the algorithm in [16]. The total number of training samples was 5943 with 180 categories. These samples are published with our dataset. This number is not plenty when constructing a classifier from scratch, but we believe it is enough when using pre-trained CNN model and not applying fine-tuning on it. The linear SVM maximize margin between categories, and this works well on our setting to avoid overfitting.

4.2. Results and Discussion

A chef will touch and release the same object multiple times during his/her food preparation. To avoid including the images derived from the same instance in both training and test samples, we applied the cross-validation method for each observation. The annotation for some observations were imcomplete, and there are categories that appears only in one observation. We skipped to test samples from such observations and categories. As a result, the number of tested samples was 4365 with 133 categories.

Table 1 shows the evaluation result from evaluating the object recognition process using the cross-validation method. In this setting, some cooking utensils appeared only in one observation, and no training samples were available. Such

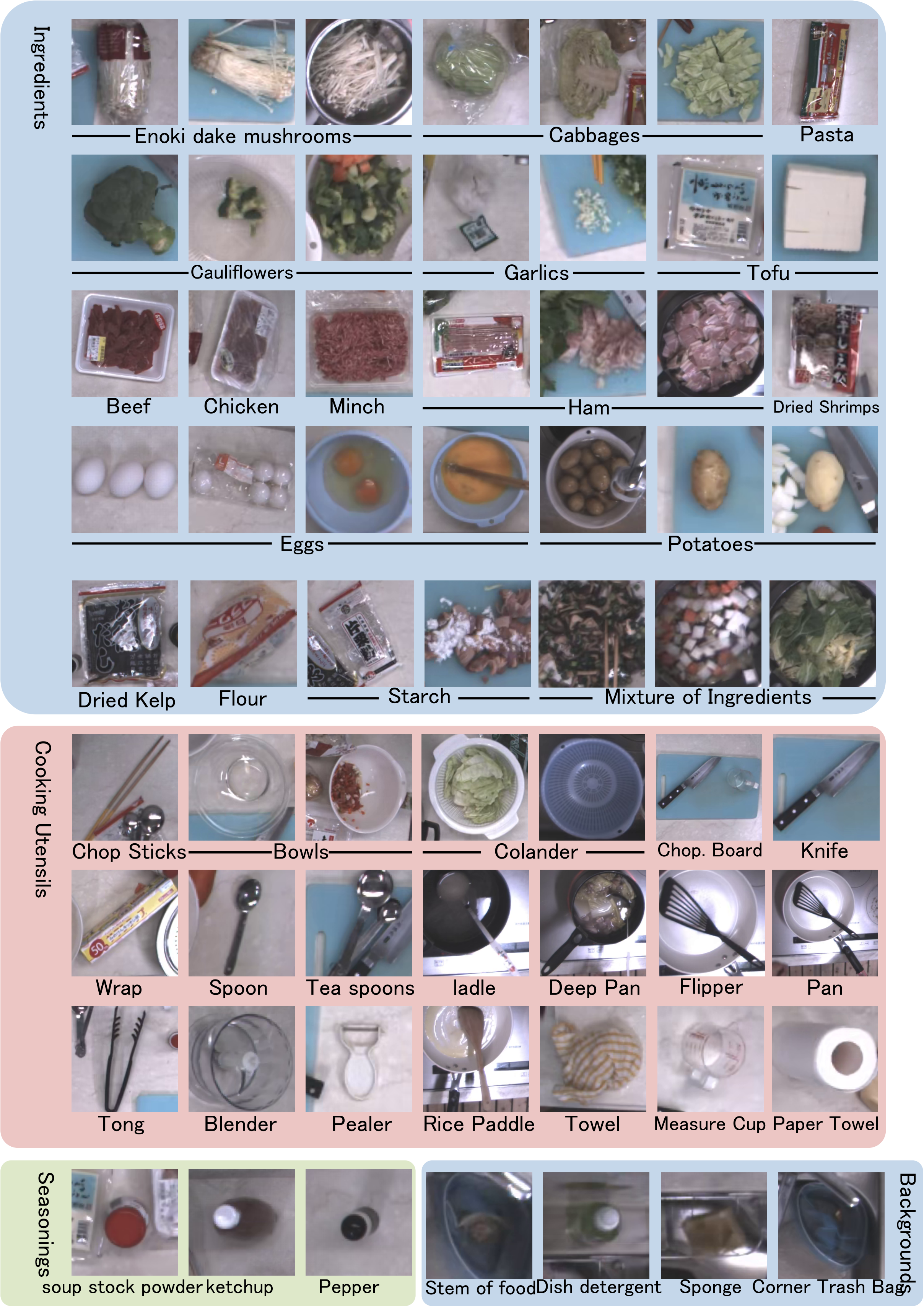

Fig. 3. Examples of image data obtained through the annotation

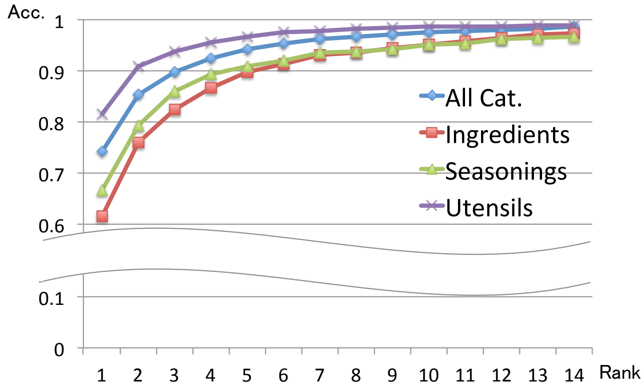

Fig. 4. CMC Curve for each category group.

objects were not counted. Inedible parts of the ingredients, bottle caps, and other objects that hardly seem to relate to the progress of food preparation are not counted in the results.

Results of utensils were more accurate than those of seasonings and ingredients despite of a larger number of classes. One reason for this is a condition of observations in the KUSK Dataset. Because all videos were observed in the same kitchen, the same instance of utensils appeared in other observations. In this sense, object recognition for utensils is nearly equal to object identification. The above situation is the same for seasonings; however, some of them have little difference when observing from the top view. The observation should be more angled for recognizing seasonings. Ingredients marked the worst accuracy in the three categories. They are observed in a variety of states, and two instances might have different appearances even when they belong to the same class. This led the lower accuracy of ingredients than other two categories.

In some cases, it is more important that the correct object class is ranked in a higher place in the recognition result. To evaluate the rank of the correct object class, we plot a cumulative match characteristic curve (CMC Curve), as shown in Fig. 4. The gradient of the CMC curve is relatively large until we get to rank 3 or 4. The accuracy at rank 5 reaches 0.896 for ingredients, 0.908 for seasonings, 0.967 for utensils, and 0.942 for all categories. This indicates that it will be much more efficient, for example, to optimize the recognition result by using contextual information from recipe text.

5. CONCLUSION

The KUSK Object Dataset is aimed at being a baseline for CV researchers who try to recognize objects on cooking counter. Additionally, we also target NLP researchers who analyze recipe texts with the observation of food preparation. Since there is a large amount of procedural texts describing food preparation, and a food preparation activity generally consists of multiple complex processes, food preparation is a practical and applicative subject for the task of video-text translation. The provided dataset is an extension of the KUSK Dataset, which has corresponding recipes and NLP results obtained from those recipes. This will activate cross-sectional researches between CV and NLP researchers for video-text translation and other applications.

The dataset is organized by two components: annotation of the records of access to objects, and the same type of data obtained automatically from multiple CV processes. The result showed the difficulty in recognizing ingredients while achieving a high accuracy at rank five on CMC curve. It is remained as a future work to recognize ingredients more accurately based on the context of food preparation, which is typically obtained from recipe texts. The recognition method using contextual information will also contribute video-text alignment and translation.

Acknowledgement

This work was supported by JSPS KAKENHI Grant Numbers 24240030, 26280084 and 16K16099.

6. REFERENCES

[1] Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth, “Every picture tells a story: Generating sentences from images,” in *ECCV 2010*, pp. 15–29. Springer, 2010.

[2] Yezhou Yang, Ching Lik Teo, Hal Daumé III, and Yiannis Aloimonos, “Corpus-guided sentence generation of natural images,” in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2011, pp. 444–454.

[3] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan, “Show and tell: A neural image caption generator,” *arXiv preprint arXiv:1411.4555*, 2014.

[4] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, and Kate Saenko, “Translating videos to natural language using deep recurrent neural networks,” *arXiv preprint arXiv:1412.4729*, 2014.

[5] Iftekhar Naim, Young Chol Song, Qiguang Liu, Henry Kautz, Jiebo Luo, and Daniel Gildea, “Unsupervised alignment of natural language instructions with video segments,” in *Proceedings of Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[6] Atsushi Hashimoto, Jin Inoue, Takuya Funatomi, and Michihiko Minoh, “How does user’s access to object make hci smooth in recipe guidance?,” in *Proc. of 6th International Conference on Cross-Cultural Design, Held as Part of HCI International 2014*, 2014, pp. 150–161.

[7] Cyrus Rashtchian, Peter Young, Micah Hodosh, and Julia Hockenmaier, “Collecting image annotations using amazon’s mechanical turk,” in *Proc. of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk*, 2010, pp. 139–147.

Table 1. Recognition Accuracy for each recipe (Cls.: # of Classes, Acc.: Accuracy, Acc. (in): Accuracy within the category)

	all categories		ingredients			seasonings			utensils		
	Cls.	Acc.	Cls.	Acc.	Acc. (in)	Cls.	Acc.	Acc. (in)	Cls.	Acc.	Acc. (in)
2014RC01	23	70.92%	6	54.64%	81.44%	5	53.70%	81.44%	11	85.25%	87.43%
2014RC02	25	79.17%	8	54.55%	63.64%	5	75.00%	63.64%	13	86.40%	87.20%
2014RC03	24	78.32%	7	78.64%	88.35%	6	61.90%	88.35%	12	82.55%	83.22%
2014RC04	20	74.79%	4	61.76%	91.18%	6	54.55%	91.18%	9	83.33%	88.89%
2014RC05	23	78.62%	6	72.41%	75.86%	4	76.47%	75.86%	14	83.91%	83.91%
2014RC06	25	64.00%	8	45.31%	56.25%	6	33.33%	56.25%	12	79.55%	79.55%
2014RC07	16	71.08%	4	54.05%	75.68%	3	83.33%	75.68%	8	78.16%	79.31%
2014RC08	14	71.05%	3	50.00%	60.00%	4	25.00%	60.00%	7	87.50%	91.67%
2014RC09	22	71.67%	6	60.34%	72.41%	5	74.29%	72.41%	12	76.52%	76.52%
2014RC10	28	78.40%	7	69.07%	82.47%	5	85.71%	82.47%	15	82.39%	84.28%
2014RC11	25	79.54%	7	60.00%	76.67%	6	82.05%	76.67%	13	85.71%	87.01%
2014RC12	25	79.77%	7	77.78%	84.44%	6	65.79%	84.44%	11	87.34%	87.34%
2014RC13	25	71.57%	6	58.33%	72.22%	5	56.00%	72.22%	15	76.12%	77.61%
2014RC14	20	68.47%	5	48.57%	71.43%	4	66.67%	71.43%	12	81.90%	84.76%
2014RC15	30	65.65%	8	49.02%	67.65%	7	55.10%	67.65%	16	73.54%	76.65%
2014RC16	21	76.89%	5	66.67%	86.11%	3	73.91%	86.11%	14	81.82%	83.77%
2014RC17	24	76.30%	7	65.67%	85.07%	5	55.00%	85.07%	13	84.82%	85.71%
2014RC18	26	72.55%	6	61.03%	77.94%	5	80.00%	77.94%	16	80.19%	81.16%
2014RC19	27	79.23%	7	62.86%	71.43%	5	72.73%	71.43%	16	84.76%	85.71%
2014RC20	15	85.84%	5	88.00%	92.00%	2	100.00%	92.00%	8	85.71%	86.90%
Total	133	74.15%	46	61.57%	77.52%	32	66.60%	77.52%	35	81.58%	83.10%

[8] Moritz Tenorth, Jan Bandouch, and Michael Beetz, “The tum kitchen data set of everyday manipulation activities for motion tracking and action recognition,” in *Proc. of ICCV Workshops*, 2009, pp. 1089–1096.

[9] F. De la Torre, J. Hodgins, J. Montano, S. Valcarcel, R. Forcada, and J. Macey, “Guide to the carnegie mellon university multimodal activity (CMU-MMAC) database,” in *Technical report CMU-RI-TR-08-22*, 2008, pp. 1–17.

[10] Atsushi Shimada, Kazuaki Kondo, Daisuke Deguchi, Géraldine Morin, and Helman Stern, “Kitchen scene context based gesture recognition: A contest in icpr2012,” in *Advances in Depth Image Analysis and Applications*, pp. 168–185. Springer, 2013.

[11] Marcus Rohrbach, Sikandar Amin, Mykhaylo Andriluka, and Bernt Schiele, “A database for fine grained activity eetection of cooking activities,” in *Proc. of 2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1194–1201.

[12] Hilde Kuehne, Ali Arslan, and Thomas Serre, “The language of actions: Recovering the syntax and semantics of goal-directed human activities,” in *Proc. of Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 780–787.

[13] S. Stein and S. J. McKenna, “Combining embedded accelerometers with computer vision for recognizing food preparation activities,” in *Proc. of UbiComp 2013*, 2013, pp. 729–738.

[14] Atsushi Hashimoto, Sasada Tetsuro, Yoko Yamakata, Shinsuke Mori, and Michihiko Minoh, “KUSK Dataset: Toward a direct understanding of recipe text and human cooking activity,” in *Workshop on Smart Technology for Cooking and Eating Activities*, 2014, pp. 583–588.

[15] Hirokuni Maeta, Tetsuro Sasada, and Shinsuke Mori, “A framework for procedural text understanding,” in *Proc. of the 14th International Conference on Parsing Technologies*, 2015.

[16] Atsushi HASHIMOTO, Takuya FUNATOMI, Kazuaki NAKAMURA, and Michihiko MINOH, “False alert rejection for detecting objects on a table by touch reasoning,” *The IEICE Trans. on Information and Systems (Japanese edetion)*, vol. 95, no. 12, pp. 2113–2123, 2012.

[17] Hidetsugu Nanba, Yoko Doi, Miho Tsujita, Toshiyuki Takezawa, and Kazutoshi Sumiya, “Construction of a cooking ontology from cooking recipes and patents,” in *Proc. of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2014, pp. 507–516.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Deep residual learning for image recognition,” *arXiv preprint arXiv:1512.03385*, 2015.