FOOD IMAGE DESCRIPTION BASED ON DEEP-BASED JOINT FOOD CATEGORY, INGREDIENT, AND COOKING METHOD RECOGNITION

Wei-Ta Chu and Jia-Hsing Lin

National Chung Cheng University, Taiwan
wtchu@ccu.edu.tw, xupruvu@gmail.com

ABSTRACT

Many works have been proposed for food image analysis, such as food recognition and ingredient recognition, in order to facilitate healthcare applications. However, relatively fewer studies have been done on jointly considering multiple factors. In this paper, we think that a food image is better described by not only what food it is but also how it was cooked. We propose neural networks to jointly consider food recognition, ingredient recognition, and cooking method recognition, and verify that recognition performance can be improved by taking multiple factors into account. We collect a food image dataset consisting of clean ingredient information, and demonstrate effectiveness of the proposed recognition models from various viewpoints.

***Index Terms*—** Joint correlation, food recognition, ingredient recognition, cooking method recognition

1. INTRODUCTION

Food image analysis has attracted much attention for its practicability and technical challenges. As more and more food images shared on social media platforms, visual food recognition is not only the foundation of healthcare-related applications, but also an important clue to explore people's living style. Previous food-related works mainly include food recognition [1][2][3][4], food quantity/calory estimation [5][6][7], and recipe retrieval [8][9].

Recently, works on image captioning emerge rapidly because of its extensive potentials in bridging the semantic gap between visual features and high-level semantics. Thanks to the rapid advancement of deep visual representation by convolutional neural network and language generation by recurrent neural network, performance of image captioning scales up to a large factor in just recent two years [10][11].

The elegant models mentioned above, however, largely focus on general-purposed image captioning, which may not well catch uniqueness of images in a specific domain. In this work, we concentrate on food image captioning for three reasons. First, tremendous amounts of food images are daily shared on social media platforms. These images not only show what a user eats, but also present the user's life style. Second, food image descriptions facilitate many valuable applications, such as health management, recipe recommendation, and restaurant recommendation. Third, unlike general image captioning, an appropriate food image description should show not only the food name, but also the way it was cooked. For example, description like "roasted beef with soft-boiled eggs" is richer than "beef and egg" when a user tries to order a meal with a menu showing food images. The verb-noun pair showing the cooking method and the ingredient makes food image description distinct from general-purposed image captioning.

In order to generate food image description consisting of verb-noun pairs (VNPs), we propose neural frameworks that jointly consider multiple factors, i.e., food name, ingredients, and cooking methods. With this joint recognition model, better recognition rate can be obtained and thus better descriptions can be generated. Contributions of this work are summarized as follows.

- A learning framework is proposed to jointly consider multiple factors as well as transfer information from one modality to another modality, in order to improve performance of a targeted task.
- Based on the proposed framework, we propose food image descriptions as a set of verb-noun pairs, generally a cooking method followed by an ingredient. Correlations between cooking methods and ingredients with given visual information are learnt based on recipe information.
- To facilitate the proposed food image description, we collect a food image dataset associated with well-organized recipe information. In contrast to previous datasets where recipe data are usually just for food recognition, we analyze cooking steps and associated ingredients, and summarize a recipe as a set of verb-noun pairs to facilitate construction of the proposed system.

The rest of this paper is organized as follows. Section 2 describes the framework generally adopted to food recognition, cooking method recognition, and ingredient recognition.

978-1-5090-6067-2/17/$31.00 ©2017 IEEE

(a) French Fries (b) Chocolate Cake (c) Chicken Curry (d) Cheese Plate (e) Caprese Salad (f) Beet Salad

Fig. 1. Sample images from the UPMC dataset.

(a) Citrus Salsa Salmon (b) Pasta (c) Maple Pear Salad (d) Cheese Cake (e) Almond Cade (f) Chocolate Whoopie Pies

Fig. 2. Sample images from our dataset.

Details of the learning framework will be provided. With these recognition results, the proposed food image description in the representation of VNPs is described in Section 3. Section 4 describes experimental results in several aspects, followed by the concluding remarks given in Section 5.

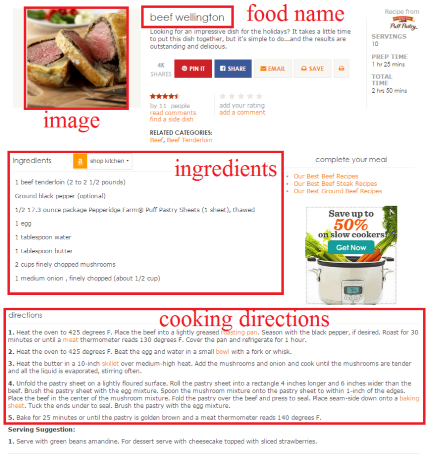
2. JOINT RECOGNITION

We advocate that recognition of one factor, e.g., food recognition, can be benefited by considering two other factors, i.e., cooking method and ingredient. In the following, we take food recognition as the main example to show how other factors are considered, while the same idea can be employed to enhance cooking recognition or ingredient recognition.

2.1. Databases

Two databases are used in this study. The first is the UPMC Food-101 dataset [8] that covers 101 food categories and includes totally 90,840 images. Images were retrieved by Google Image search, with queries from the 101 labels taken from the ETHZ Food-101 dataset [2]. In order to study recipe recognition, raw HTML pages that embed these images were also collected. The number of images having corresponding HTML text is 86,574. Figure 1 shows several sample images and corresponding food names from the UPMC dataset. Some images are with complex background, e.g., the Cheese Plate in Figure 1(d), and accurately recognizing them is not an easy task.

The UPMC Food-101 dataset is quite challenging because images were collected from uncontrolled sources. In addition, the collected HTML pages may not highly relate to the embedded food images. To facilitate more precise food image description, we need a dataset with clean recipe information. For this purpose, we crawl images of ten food categories defined in Recipe.com[1], including *beef, bread, burger, cake, casseroles, chicken, chili, cookies, fruit, and grilling*. We collected 9,363 images in total, with each image associated with a clean recipe. Figure 2 shows several sample images and the corresponding food names from our dataset. Figure 3 shows a food called *Beef Wellington* and the corresponding recipe consisting of ingredients and cooking directions. This example also shows the shortage of food name for us to realize what this food really is. From the food name, we only know it is a kind of beef meal, but have no idea about how it was cooked and hardly imagine how it tastes. Associating cooking methods and ingredients is thus important in food image description.



beef wellington

Looking for an impressive dish for the holidays? It takes a little time to put this dish together, but it's simple to do...and the results are outstanding and delicious.

SERVINGS
10

PREP TIME
1 hr 25 mins

TOTAL TIME
2 hrs 50 mins

by 11 people
read comments
find a side dish

add your rating
add a comment

RELATED CATEGORIES:
Beef, Beef Tenderloin

ingredients

1 beef tenderloin (2 to 2 1/2 pounds)

Ground black pepper (optional)

1/2 17.3 ounce package Pepperidge Farm® Puff Pastry Sheets (1 sheet), thawed

1 egg

1 tablespoon water

1 tablespoon butter

2 cups finely chopped mushrooms

1 medium onion , finely chopped (about 1/2 cup)

complete your meal

- Our Best Beef Recipes
- Our Best Beef Steak Recipes
- Our Best Ground Beef Recipes

directions

1. Heat the oven to 425 degrees F. Place the beef into a lightly greased roasting pan. Season with the black pepper, if desired. Roast for 30 minutes or until a meat thermometer reads 130 degrees F. Cover the pan and refrigerate for 1 hour.

2. Heat the oven to 425 degrees F. Beat the egg and water in a small bowl with a fork or whisk.

3. Heat the butter in a 10-inch skillet over medium-high heat. Add the mushrooms and onion and cook until the mushrooms are tender and all the liquid is evaporated, stirring often.

4. Unfold the pastry sheet on a lightly floured surface. Roll the pastry sheet into a rectangle 4 inches longer and 6 inches wider than the beef. Brush the pastry sheet with the egg mixture. Spoon the mushroom mixture onto the pastry sheet to within 1-inch of the edges. Place the beef in the center of the mushroom mixture. Fold the pastry over the beef and press to seal. Place seam-side down onto a baking sheet. Tuck the ends under to seal. Brush the pastry with the egg mixture.

5. Bake for 25 minutes or until the pastry is golden brown and a meat thermometer reads 140 degrees F.

Serving Suggestion:

1. Serve with green beans amandine. For dessert serve with cheesecake topped with sliced strawberries.

Fig. 3. A sample food image (*Beef Wellington*) and its corresponding ingredients and cooking directions in our dataset.

[1] http://www.recipe.com

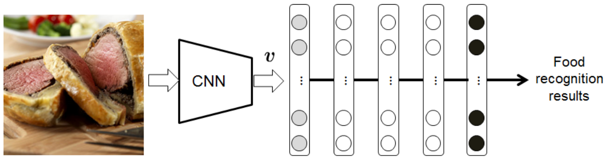


Fig. 4. The baseline deep framework for food recognition.

2.2. Baseline Model

Food Recognition. Figure 4 shows the framework of a baseline model for food recognition. Given a food image, we extract its convolutional neural network (CNN) features based on the MatConvNet toolbox [12] with the vgg-f pretrained model [13]. Results of the seventh layer, i.e., the last fully-connect layer, are taken as the image representation, which is 4,096-dimensional and is shown as $\boldsymbol{v}$ in Figure 4. Based on this image representation, a five-layer fully-connected neural network is constructed to do recognition. The input layer consists of 4,096 nodes, followed by three fully-connected layers with 2,048, 4,096, and 2,048 nodes, respectively. The output layer is a softmax layer with 101 nodes, which outputs the probabilities of the given image belonging to 101 food classes defined in the UPMC dataset.

To train this network, from each food class of the UPMC dataset we randomly select 500 images as the training data, and the remaining images are used for testing. The loss function to be minimized is cross entropy, and the optimizer is Adam. In the experiment, we perform training and testing based on the random-split scheme for five times, and report the average recognition result.

Ingredient Recognition. The same framework as shown in Figure 4 is also adopted to do ingredient recognition. We build the ingredient recognition model based on our dataset, because it has clean recipe information. For ingredient information, we manually filter out stop words and commonly-used units like spoon and jar. Finally 130 different ingredients are retained in total. Unlike the one-hot representation in food recognition, the ground truth vector $\boldsymbol{y}$ is a 130-dimensional binary vector where multiple entries would be unity because a food often contain multiple ingredients.

Cooking Method Recognition. Based on the cooking direction information of our dataset, we conclude ten common actions: *cook, bake, toast, roast, season, grill, broil, heat, simmer, and stew*. A food may be prepared with several actions, like *roast the beef, and cook the mushrooms*. The ground truth vector is thus a 10-dimensional binary vector where multiple entries would be unity. The same framework as shown in Figure 4 is also adopted for cooking method recognition.

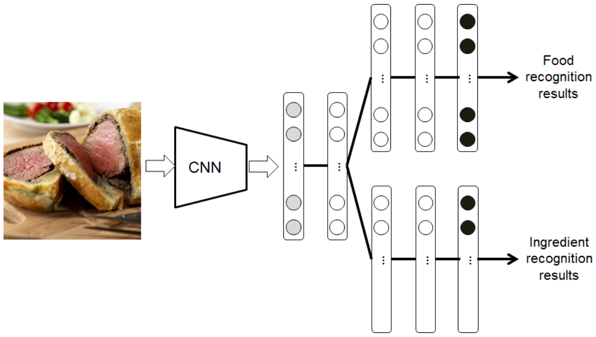


Fig. 5. Illustration of the first joint model, with the idea of similar mid representations.

2.3. Enhanced Models Considering Correlation

To enhance performance of food image analysis, we conjecture that food name, ingredients, and cooking methods are correlated, and by jointly considering multiple factors, performance gain can be obtained. Similar idea was also recently proposed in [14] and [9]. In [14], ingredients were first implicitly detected based on part-based texture features, and food classification was then achieved by considering results of ingredient recognition with a multikernel support vector machine. In [9], end-to-end deep networks with variations of shared layers and classification layers were proposed. Our work is conceptually similar to [9], with more consideration on cooking method recognition.

Joint Model 1: Shared Mid Representation. The first idea to boost food recognition is that the mid representation learnt for three recognition tasks should be similar. Figure 5 shows the framework of the first joint model. To make the figure simple and clear, we just illustrate the case where ingredient recognition and food recognition are jointly considered.

The cost function to construct the joint model is defined as the sum of two types of cross entropy values: $C = \lambda E_f(\boldsymbol{y}_f, \hat{\boldsymbol{y}}_f) + (1-\lambda)E_i(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i)$, where $E(\boldsymbol{y}, \hat{\boldsymbol{y}})$ is the cross entropy between $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$, $\boldsymbol{y}_f$ and $\boldsymbol{y}_i$ are the ground truth vectors of food and ingredient, respectively, and $\hat{\boldsymbol{y}}_f$ and $\hat{\boldsymbol{y}}_i$ are the output probability vectors of food names and ingredient, respectively. The parameter λ is used to weight the importance of food recognition and ingredient recognition, and is set as 0.5 currently.

Joint Model 2: Early Fusion. The second idea is simply concatenating the probability vector of ingredient recognition with the CNN feature of the given image to form an *enhanced* image descriptor. This idea is similar to early fusion widely used in integrating multimodal features. Figure 6 illustrates this idea. Note again that, to make the figure simple and clear, we only illustrate using ingredient recognition results to enhance food recognition here. Based on such concatenated descriptor, the cross entropy between predicted results and ground truth is minimized to train the neural network.

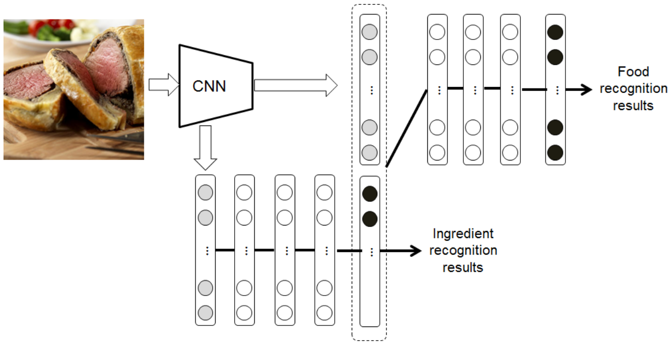


Fig. 6. Illustration of the second joint model, with the idea of early fusion.

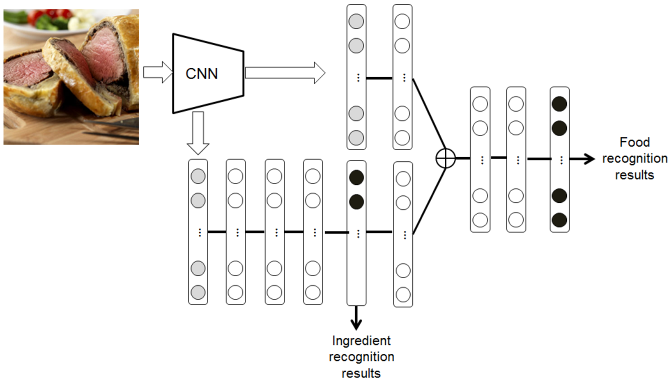


Fig. 7. Illustration of the third joint model, with the idea of transformed fusion.

Joint Model 3: Transformed Fusion. In joint model 2, we concatenate the probability vector of ingredient recognition with the CNN feature, which in fact have different physical meanings. On the other hand, in joint model 3 we propose to transform both vectors by a hidden layer into a common space, and then combine the transformed vectors by adding them together. Figure 7 illustrates this idea. We again use cross entropy as the loss function to train the neural network.

Note that the proposed joint models can be adopted to improve ingredient recognition based on results of food recognition and cooking method recognition, or improve cooking method recognition based on results of food recognition and ingredient recognition. In the evaluation section, we will compare recognition performance of the baseline model and three joint models.

3. FOOD IMAGE DESCRIPTION

Given a test image I, the cooking method recognition model mentioned above would output a probability vector $\boldsymbol{c} = (p_1^c, p_2^c, ..., p_M^c)$ showing the probability of I being cooked by each method. Similarly, the ingredient recognition method would output a probability vector $\boldsymbol{g} = (p_1^g, p_2^g, ..., p_N^g)$ showing the probability of I containing each ingredient. Conceptually, the most probably verb-noun pair can be found by finding (c^*, g^*) such that

$$(c^*, g^*) = \arg \max_{\substack{i=1,...,M \\ j=1,...,N}} p_i^c \times p_j^g. \tag{1}$$

To further consider the correlation between cooking methods and ingredients, we process the collected recipes by pairing each detected cooking term, e.g., roast, with its closest succeeding ingredient, e.g., beef. We thus found a large number of verb-noun pairs, like *roast beef* and *cook tomato*, from the recipe set. The frequency of each VNP is then normalized to be the prior probability of an action taken to cook an ingredient. The prior probability of the cooking method i used to cook the ingredient j is denoted as p_{ij}^r. With this information, the most probably VNP is then determined by finding (c^*, g^*) such that

$$(c^*, g^*) = \arg \max_{\substack{i=1,...,M \\ j=1,...,N}} p_{ij}^r \times p_i^c \times p_j^g. \tag{2}$$

In the evaluation, we in fact find the five VNPs with five largest probabilities to be image descriptions.

4. EXPERIMENTAL RESULTS

4.1. Performance of Joint Models

Table 1. Accuracy of food recognition based on the baseline model, the three joint models enhanced by ingredient, cooking, or both.

	with ingredient	with cooking	with both
Joint Model 1	0.3885	0.3983	0.0106
Joint Model 2	0.3884	**0.4044**	0.3943
Joint Model 3	0.3976	0.4012	0.3995
Baseline	0.3833		
[8] (visual only)	0.3391		

We evaluate performance of the proposed joint models based on the UPMC dataset. To more finely evaluate the influence of different factors, we evaluate three joint models enhanced by different combinations of factors. Table 1 shows performance of food recognition obtained based on different models. As can be seen, all three joint models consistently outperform the baseline model, except for the first joint model considering both ingredient and cooking. Although results of ingredient recognition and cooking method recognition are not perfect, the joint models take advantage of extra information to facilitate more accurate food recognition. The second observation is that jointly considering more information does not necessarily yield better performance. In Table 1, the best performance is obtained by the early fusion method (Joint Model 2) that fuses the result of cooking method recognition with the CNN visual descriptor. For the UPMC dataset

Table 2. Accuracy of ingredient recognition based on the baseline model, the three joint models enhanced by food, cooking, or both.

	with food	with cooking	with both
Joint Model 1	0.3700	0.3620	0.5552
Joint Model 2	0.5550	0.5550	0.5549
Joint Model 3	**0.5552**	0.5445	0.5358
Baseline	0.5379		

Table 3. Accuracy of cooking method recognition based on the baseline model, the two joint models enhanced by food, cooking, or both.

	with food	with ingredient	with both
Joint Model 1	0.5550	0.5550	0.5549
Joint Model 2	0.5496	0.5496	0.5496
Joint Model 3	**0.5558**	0.5476	0.5480
Baseline	0.5498		

that includes 101 food classes, we obtain 40.44% recognition accuracy, which significantly outperforms 33.91% accuracy reported in [8].

Table 2 shows recognition accuracy of ingredients based on different models. Here we again see superior performance of joint models. More interestingly, we observe that the first joint model does not yield reliable performance in ingredient recognition. The best performance is obtained by the third joint model, with the help of food recognition results. The performance difference between the second and the third joint models is slight. Jointly considering more information doesn't yield better performance.

Table 3 shows recognition accuracy of ingredients based on different models. We observe that the first joint model consistently outperforms the baseline model. Through the results mentioned above, we verify effectiveness of joint models, and conclude that joining appropriate information is the key to get performance gain.

4.2. Performance of Food Image Description

4.2.1. Effectiveness of VNPs

Because there is no ground truth of food image description, we evaluate the proposed food image description based on subjective tests. Ten subjects were invited to join the experiment, where each person was randomly given fifteen to twenty food images with the automatically generated descriptions. Two types of descriptions were generated for food images randomly selected in the test set of the UPMC database: (1) only ingredient recognition results; and (2) the generated verb-noun pairs. These two types of descriptions were randomly juxtaposed, and the subjects were asked to select which one was better to annotate the given food image.

Overall, VNP is viewed better in 86 of 155 food images, while the rest 69 of 155 food images are viewed to be better annotated by ingredient only. The VNP-based description is not significantly better because both results of ingredient recognition and cooking method recognition are still not good enough (see Table 2 and Table 3). The effectiveness of VNPs might be largely elevated if these two recognition results can be improved.

Table 4. Performance of food image captioning, in the representation of the number of images that are viewed to have better caption results.

Dataset	CaptionBot	Our method
UPMC (totally 58 images)	19	39
Our dataset (totally 100 images)	16	84

4.2.2. General-Purposed Image Captioning vs. VNPs

We design a subjective experiment to compare results of general-purposed image captioning with food-specific VNPs. Given a food image selected in the test set of the UPMC database, or our dataset, we generate VNPs by our method and generate a general-purposed image caption by Microsoft CaptionBot service[2]. Two types of descriptions were presented to subjects, who were then asked to measure which caption is better to describe the given food image. As can be seen in Table 4, among the 58 test images in the UPMC dataset, our VNP-based captions are viewed to be better description in 39 images. Among the 100 test images in our dataset, the VNP-based captions are viewed to be better description in 84 images. Results for images from our dataset are much better than that from the UPMC dataset. The main reason is that the prior probability described in eqn. (2) can be more accurately estimated in our dataset, because our dataset have cleaner recipe information.

Fig. 8 shows a food image representing *Burger*, and Table 5 shows recognition results and image descriptions in the representation of VNPs as well as the image caption generated by CaptionBot. As can be seen, we correctly recognize that the given image is Burger. The top five recognized ingredients are egg, milk, cheese, corn, and lemon, which are satisfactory results. The top three recognized cooking methods are heat, bake, and cook, which can be imagined to make a burger. By finding the most likely VNPs, the top five VNPs are bake egg, bake onion, bake tomato, bake beef, and bake cheese. All of them are appropriate VNPs because they are all necessary processes in making a burger. By showing food recognition result followed by VNPs, we can generate a description that provides information richer than a general-purposed image caption, as shown in the last row of Table 5.

[2] https://www.captionbot.ai/

Table 5. Recognition and image caption (VNP) results corresponding to Fig. 8.

Types	Results
Food recognition	Burger
Ingredient recognition (top 5)	egg, milk, cheese, corn, lemon
Cooking method recognition (top 3)	heat, bake, cook
Image captioning (food name followed by top 5 VNPs)	Burger: bake with egg, bake with onion, bake with tomato, bake beef, bake with cheese
Microsoft CaptionBot	I think it's a sandwich on a plate.

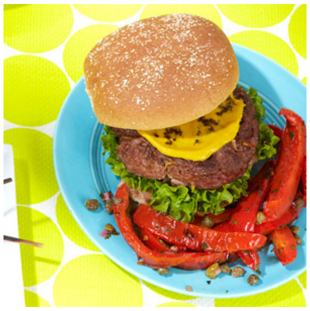


Fig. 8. A food image representing *Burger*.

5. CONCLUSION

We have presented a food image description system based on joint recognition. Three schemes are proposed to join information from multiple factors in a learning framework. Based on recognition results, we generate verb-noun pairs that not only shows what food it is but also show how it was cooked. In the evaluation, we verify the effectiveness of joint models, and show that VNPs are more effective in describing food images, as compared to general-purposed image captioning. In the future, how to more tightly integrate different recognition results or intermediate representation in a learning framework is still an important issue.

Acknowledgement. The work was partially supported by the Ministry of Science and Technology of Taiwan under the grant MOST105-2628-E-194-001-MY2, MOST104-2221-E-194-014, and MOST103-2221-E-194-027-MY3.

6. REFERENCES

[1] S. Yang, M. Chen, D. Pomerleau, and R. Sukthankar, "Food recognition using statistics of pairwise local features," *Proc. of CVPR*, pp. 2249–2256, 2010.

[2] L. Bossard, M. Guillaumin, and L. Van Gool, "Food-101 – mining discriminative components with random forests," *Proc. of ECCV*, 2014.

[3] H. Kagaya, K. Aizawa, and M. Ogawa, "Food detection and recognition using convolutional neural network," *Proc. of ACM MM*, pp. 1085–1088, 2014.

[4] Y. Kawano and K. Yanai, "Foodcam: A real-time food recognition system on a smartphone," *MTAP*, vol. 74, no. 14, pp. 5263–5287, 2015.

[5] F. Zhu, M. Bosch, I. Woo, S. Kim, C. J. Boushey, D. S. Ebert, and E. J. Delp, "The use of mobile devices in aiding dietary assessment and evaluation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 4, pp. 756–766, 2010.

[6] A. Myers, N. Johnston, V. Rathod, A. Korattikara, A. N. Gorban, N. Silberman, S. Guadarrama, G. Papandreou, J. Huang, and K.Murphy, "Im2calories: Towards an automated mobile vision food diary," *Proc. of ICCV*, pp. 1233–1241, 2015.

[7] K. Aizawa, Y. Maruyama, H. Li, and C. Morikawa, "Food balance estimation by using personal dietary tendencies in a multimedia food log," *IEEE TMM*, vol. 15, no. 8, pp. 2176–2185, 2013.

[8] X. Wang, D. Kumar, N. Thome, M. Cord, and F. Precioso, "Recipe recognition with large multimodal food dataset," *Proc. of ICME Workshop*, 2015.

[9] J. Chen and C.-W. Ngo, "Deep-based ingredient recognition for cooking recipe retrieval," *Proc. of ACM MM*, pp. 32–41, 2016.

[10] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," *Proc. of CVPR*, pp. 3128–3137, 2015.

[11] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," *Proc. of CVPR*, pp. 3156–3164, 2015.

[12] A. Vedaldi and K. Lenc, "Matconvnet: Convolutional neural networks for matlab," *Proc. of ACM MM*, pp. 689–692, 2015.

[13] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *Proc. of BMVC*, 2014.

[14] H. He, F. Kong, and J. Tan, "Dietcam: Multiview food recognition using a multikernel svm," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 3, pp. 848–855, 2016.