Image-Based Food Calorie Estimation Using Recipe Information

SUMMARY  Recently, mobile applications for recording everyday meals draw much attention for self dietary. However, most of the applications return food calorie values simply associated with the estimated food categories, or need for users to indicate the rough amount of foods manually. In fact, it has not been achieved to estimate food calorie from a food photo with practical accuracy, and it remains an unsolved problem. Then, in this paper, we propose estimating food calorie from a food photo by simultaneous learning of food calories, categories, ingredients and cooking directions using deep learning. Since there exists a strong correlation between food calories and food categories, ingredients and cooking directions information in general, we expect that simultaneous training of them brings performance boosting compared to independent single training. To this end, we use a multi-task CNN. In addition, in this research, we construct two kinds of datasets that is a dataset of calorie-annotated recipe collected from Japanese recipe sites on the Web and a dataset collected from an American recipe site. In the experiments, we trained both multi-task and single-task CNNs, and compared them. As a result, a multi-task CNN achieved the better performance on both food category estimation and food calorie estimation than single-task CNNs. For the Japanese recipe dataset, by introducing a multi-task CNN, 0.039 were improved on the correlation coefficient, while for the American recipe dataset, 0.090 were raised compared to the result by the single-task CNN. In addition, we showed that the proposed multi-task CNN based method outperformed search-based methods proposed before.

key words:  food image recognition, image-based food calorie estimation, convolutional neural network, multi-task CNN

1. Introduction

In recent years, because of a rise in health thinking on eating, many mobile applications for recording everyday meals have been released so far. Some of them employ food image recognition which can estimate not only food names but also food calories[1]–[6]. However, in most of the cases, the estimated calories are just associated with the estimated food categories, or the relative size compared to the standard size of each food category which is usually indicated by a user manually. Currently, no applications which can estimate food calories automatically exist. Although most of the image recognition tasks including food category recognition have been almost solved due to great progress of CNN-based image recognition methods, fully-automatic food calorie estimation from a food photo has still remained an unsolved problem. We think that food calorie estimation not only helps people’s health a lot, but also is promising as a new problem of image recognition studies.

Regarding food calorie estimation, a lot of approaches have been proposed so far. The main approach is to estimate calories based on the estimated food category and its size or volume, which is a quite standard approach [1], [2], [4], [5], [7], [8]. Since food calories strongly depend on food categories and volumes, this approach is effective and important.

The other approach is to estimate calories from food photos directly not taking account of food categories and volumes. The works adopting this approach are a few [9]. Food calories strongly depend on the food categories, volumes, ingredients and cooking directions. Even if food categories are the same, the food calories are different depending on used ingredients and cooking directions as shown in Fig. 1. Therefore, we think estimating calories from appearance is important in the task of food calorie estimation, which cannot be resolved by only food category estimation. Estimating food calories directly from a food photo potentially make it possible to account for the intra-category differences as shown in Fig. 1. The biggest problem on this approach was that it is difficult to prepare calorie-annotated food image datasets. Since Miyazaki et al. used non-public photos gathered in a commercial service of food photo analysis, FoodLog, they were not able to make their dataset open to the public. No other works than [9] employing direct approaches have been published so far.

Recently, many cooking recipe sites such as CookPad

Fig. 1  The differences of food calorie values within the same food categories. “Spaghetti” in the top row and “Miso soup” in the bottom row are shown, respectively.
and AllRecipes exists. They provide cooking recipe information consisting of meal photos, lists of ingredients, and texts of cooking directions. Some of them have cooking recipes with calorie values as well. Although the ratio of calorie-annotated recipes is not high, the total numbers of recipes on the Web is very large. For example, Cooc-Pad has announced that it has more than two million cooking recipes. In fact, we successfully have gathered 83,000 calorie-annotated Japanese food recipes and 24,000 calorie-annotated American food recipes in the experiments.

In this work, basically we adopt the latter approach which is estimating food calories directly by taking advantage of large calorie-annotated recipes on the Web, and propose simultaneous learning of food calories, categories, ingredients and cooking directions for the food calorie estimation from a food photo. Since there exists a strong correlation between food calories and food categories, ingredients and cooking directions information, we expect that simultaneous training of them brings performance boosting compared to independent single training. To this end, we use a multi-task CNN [10]. Chen and Ngo [11] proposed using a multi-task CNN to estimate food categories and food ingredients at the same time, and proved that simultaneous estimation boosted estimation performance on both tasks. Inspired by their work, we introduce a multi-task CNN for food calorie estimation.

The food calorie estimation is treated as a regression problem where the input is a food photo and the output is the value of a food calorie. In this paper, we assume that a given food image contains only one dish and the goal is estimating the value of food calories for one person as shown in Fig. 1. The food category estimation is treated as a normal classification problem. Regarding food ingredient estimation, we convert food ingredients information into real value vectors by Word2Vec [12], and train a CNN to estimate a vector of food ingredient information. Regarding cooking direction estimation, sentences of the cooking directions are converted into real value vectors, and we train a CNN to estimate a vector of cooking direction information.

In addition, in this paper we collect calorie-annotated recipe data from the online cooking recipe sites, and constructed two kinds of datasets, Japanese recipe dataset and American recipe dataset. Although food photo datasets such as Food-101 [13], UEC Food-100 [14] and VIREO Food-172 [11] have been published so far, no food photo datasets have calorie annotation.

To summarize our contributions in this paper, we (1) propose to use multi-task CNN for the task of food calorie estimation with simultaneous learning of food calories, categories, ingredients and cooking directions, and (2) construct calorie-annotated food photo datasets by collecting recipe data from online cooking recipe sites and (3) show effectiveness of the multi-task CNN based approach with the comprehensive experiments including comparison to a search-based baseline method. Note that this paper is based on our previous conference papers [15], [16] both of which are available on the authors’ lab HP†.

2. Related Work

2.1 Image-Based Food CalorieEstimation

Various approaches has been proposed so far and the main approach is to estimate calories based on estimated food categories and its size or volume using the value of food calorie per unit area or volume.

Chen et al. [7] proposed an image-based food calorie estimation method that estimates food categories and volumes by depth cameras such as Kinect. Since depth cameras such as Kinect are not so common at present, it is difficult for ordinary people to use regularly in usual eating situations.

Kong et al. [5] proposed a mobile application to estimate food calories from images multiple images, “DietCam”. They carried out segmentation and food item recognition, and in addition reconstructed 3D volumes of food items and calculate food calories based estimated volumes. 3D reconstruction was performed with SIFT-based keypoint matching and homography estimation which were a standard method of 3D stereo-vision. Also, Dehais et al. [4] carried out detection of dishes, segmentation, food categorization. Furthermore, 3D reconstruction with images from multiple viewpoints is performed, finally estimate the amount of carbohydrates.

Pouladzadhe et al. [8] proposed a food calorie estimation system which needed two dish images taken from the top and the side and used a thumb of a user as a reference object. Their method to estimate volumes were calculated by multiplying the size of food items estimated from the top-view image by the height estimated from the side-view image, which was relatively a straight-forward way.

The problem of these volume-based approach is that multiple-view food photos annotated with calorie values are required for evaluation, which are very hard to obtain. For this reason, all the works based on this approach uses in-house small-scale data sets in which the number of food photos and food categories are very limited.

Myers et al. [1] proposed “Im2Calories” which was a food calorie estimation application for Android smartphones. Although this projects were carried out by the Google Research and they made the press-release on this project in 2015, Android applications as well as the dataset which they announced they would make to the public in their paper have not been released yet. They employed state-of-the-art CNN-based segmentation [17] and CNN-based 3D volume estimation from 2D single images [18] in addition to CNN-based food category recognition. However, the experiments described in the paper was small-scale, which seemed far from practical use for common consumers. Okamoto et al. [2] proposed an image-based calorie

†We added new experiments on comparison to search-based methods to Sect. 5.5 over these conference papers.
estimation system which estimated food calories automatically by simply taking a meal photo from the top with a pre-registered reference object.

As described above, estimating food categories and volumes is a standard approach for estimating food calories from the food photo. In contrast to this, Miyazaki et al. [9] estimated calories from food photos directly without estimating food categories and volumes. The biggest difficulty on direct calorie estimation is creating datasets which contains calorie-annotated food images. They hired dietitians to annotate calories on 6512 food photos which up-loaded to the commercial food logging service, Food-Log$. Unfortunately, their dataset was not released, because they used the food photos picked up from the master image database of the commercial service. In their work, they adopted image-search based calorie estimation, in which they searched the calorie-annotated food photo database for the top k similar images based on conventional hand-crafted features such as SURF-based BoF and color histograms and estimated food calories by averaging the food calories of the top k food photos. Since their method ignored information on food categories, their method was applicable for any kinds of foods. However, the number of food images was not enough for the search-based method, and the employed image features was too simple. As results, they failed to estimate food calories with high accuracy. On the other hand, because in our work we use CNN which is successful in image recognition, we expect great improvement in terms of accuracy.

2.2 Multi-Task CNNs

To learn multiple tasks simultaneously, multi-task CNN has been proposed so far [10]. In the original work, it was applied to the face attribute detection task.

Recently, it was applied to food category and ingredient estimation by Chen and Ngo [11]. They showed that simultaneous estimation boosted estimation performance on both tasks. Inspired by their work, we introduce a multi-task CNN for simultaneous learning of food calories, categories, ingredients and cooking directions.

3. Method

The food calorie estimation is treated as a regression problem where the input is a food photo and the output is the value of a food calorie. We assume that a given food image contains only one dish and the output is the value of a food calorie for one person. Regarding food ingredient estimation, we convert the food ingredient information to a real number vector by Word2Vec [12], and use the vectors for training of the CNNs. Also, in the cooking direction estimation, sentences of cooking directions are converted into a real number vector.

Fig. 2 Overview of our multi-task CNN.

3.1 Overview of Multi-Task CNN

The architecture of our multi-task CNN is based on VGG-16 [19]. As shown in Fig. 2, the fully-connected layer (fc6) is shared by all tasks, and the fc7 layer is branched to each task, so that each task has the fc7 layer and the output layer (fc8) independently. Chen and Ngo [11] showed that the best multi-task CNN architecture for food recognition which is based on VGG-16 is one having one shared fc layer and two individual fc layers. We follow this architecture in our work on multi-task food calorie estimation.

In this paper, we train food calories, categories, ingredients and cooking directions simultaneously. Let $L_{cal}$, $L_{cat}$, $L_{ing}$, $L_{dir}$ be the loss functions of these four task. The overall loss function $L$ is represented as follows:

$$L = L_{cal} + \lambda_{cal} L_{cat} + \lambda_{ing} L_{ing} + \lambda_{dir} L_{dir}$$ (1)

We denote $L_{ab}$ as an absolute error and $L_{re}$ as a relative error, $L_{cal}$ is defined as follows:

$$L_{cal} = \lambda_{re} L_{re} + \lambda_{ab} L_{ab},$$ (2)

where $\lambda_{re}$, $\lambda_{ab}$, $\lambda_{cat}$, $\lambda_{ing}$ and $\lambda_{dir}$ are the weight on the loss functions, and the value of each $\lambda$ is determined so that all loss terms are equally treated. Details are described in the section of the experiments.

3.2 Food Calorie Estimation

The food calorie estimation task has the fc7a layer with 4096 dimension and an output layer (fc8a) composed of one unit which outputs the food calorie. Because food calories are real values, this task is treated as a regression problem. Generally, in the regression problem, a mean square error is used as the loss function, although in this paper we use the loss function of Eq. (2). The absolute error is the absolute value of the difference between the estimated value and the ground-truth, and the relative error is the ratio of the absolute error to ground-truth. Since both errors are important indicators, we think that it is desirable to consider both. Combining absolute error and relative error as in Eq. (2), both errors decrease in training. Let $y$ be the estimated value of an image $x$ and $g$ be the ground-truth, $L_{ab}$ and $L_{re}$ are defined as following:

$$L_{ab} = |y - g|$$ (3)
3.3 Food Category Estimation

The branch for food category estimation has the fc7b layer with 4096 dimension and an output layer (fc8b) the number of the elements of which equals to the number of food categories. To obtain category probability, we use standard soft-max function for an activate function of fc8b. Therefore, we use a soft-max cross entropy function as a loss function for this branch. Let \( y_k \) be the value of the \( k \)-th element of fc8b and \( g_k \) be the \( k \)-th element of ground-truth one-hot vectors. \( L_{cat} \) are defined as following:

\[
L_{cat} = - \sum_{k=1}^{n} g_k \log y_k, \tag{5}
\]

where \( n \) represents the number of food categories.

3.4 Food Ingredient Estimation

In this paper, we convert each word of food ingredient names into a real-value vector by Word2Vec [12]. Since each recipe contains multiple ingredients, we obtain a vector of ingredients information for each recipe by calculating a weighted linear combination of Word2Vec vectors of all the ingredient words for each recipe. We use this calculated vector as a representation of ingredient information on food recipes. In case of using ingredient vectors as training data, it is difficult to recognize ingredients individually for a given food image. This is not a big problem, since our objective to introduce ingredient information is not the ingredients recognition. We expect to obtain the effect of simultaneous learning by multi-task CNNs and to improve estimation accuracy of food calorie values and categories. Therefore we adopt this method for training food ingredients information.

In this paper we use the model of Word2Vec pre-trained with a large-scale recipe corpus. For the Japanese recipe dataset, we use sentences of cooking directions in the CookPad recipe dataset, while for the American recipe dataset, we use sentences of cooking directions in American recipe dataset described in Sect. 4.2. The sentences used for training in Word2Vec are pre-processed such as removal of low frequency words and sub-sampling of high frequency words. We use Skip-gram [12] as model and perform negative sampling [12] for training of Word2Vec.

For each recipe data, we use only the words of food ingredient name that is top \( N_{max} \) of the value of the TF-IDF. In the experiments, We set \( N_{max} \) to the average number of food ingredient words of each recipe. Finally, a food ingredient vector for each recipe data is calculated from the vectors obtained from Word2Vec and the values of TF-IDF. Let \( w_i \) as the words of food ingredient name at recipe data \( r_j \), food ingredient vector \( v_j \) of recipe data \( r_j \) are defined as following:

\[
v_j = \sum_{k=1}^{N} tfidf_{i,j} \ast \text{word2vec}(w_k) \tag{6}
\]

\( N \) is the number of words used in each recipe data. \( \text{word2vec}(w_k) \) is a real number vector of \( w_k \) obtained from Word2Vec and \( tfidf_{i,j} \) is the value of TF-IDF of \( w_k \) at recipe data \( r_j \). Then \( v_j \) is L2 normalized.

Training of food ingredients information is realized as a task of estimating of a food ingredient vector. This food ingredient estimation task has the fc7c layer of 4096 dimensions and an output layer (fc8c) composed of units of dimensions of food ingredient vector. Let \( y_i \) as the output of unit \( i \) and \( g_i \) as the ground-truth, \( L_{ing} \) are defined as following:

\[
L_{ing} = \frac{1}{2} \sum_{k=1}^{n} (g_k - y_k)^2 \tag{7}
\]

3.5 Cooking Directions Estimation

In addition to ingredient information, we use cooking directions as addition auxiliary information for multi-task learning as well.

In the same way as ingredient information, we convert each word in the sentences of cooking directions into a real-value vector by Word2Vec [12], and calculate a weighted linear combination of them for each recipe.

To obtain cooking direction vectors, we only use nouns, verbs and adjectives in the sentences of cooking directions, and use words with high TF-IDF values. For each recipe data, we use only the words of sentences of cooking directions that are top \( N_{max} \) of the value of the TF-IDF. In the experiments, We set \( N_{max} \) to the average number of words included in cooking directions of each recipe. Finally, cooking direction vector for each recipe data is calculated from the vectors obtained from Word2Vec with the TF-IDF weights. Let \( w_i \) be the words of sentences of cooking directions at recipe data \( r_j \). Cooking direction vector \( v_j \) of recipe data \( r_j \) is calculated by Eq. (6). Training of cooking direction information is realized as a task of estimating this cooking directions vector. This cooking direction estimation task has the fc7d layer of 4096 dimensions and an output layer (fc8d) composed of units of dimensions of cooking directions vector. \( L_{ing} \) is defined as Eq. (7).

4. Construction of Calorie-Annotated Food Photo Dataset

As far as we know, at present, there exists no publicly available food image dataset annotated with food calorie values. It costs too much to create calorie-annotated food image dataset by hand. Instead, we focus on collecting such data from the Web. In fact, some commercial cooking recipe sites provide recipes annotated with calorie values. In addition, they provide information on a food ingredient list and a description of cooking direction for each recipe as well. In this paper, we collect such information from some commercial
Web sites, and create recipe datasets annotated with calorie values. In order to confirm effectiveness of multi-task training for food calorie estimation, we construct two kinds of datasets on Japanese and American foods, and use both of them in the experiments.

4.1 Japanese Calorie-Annotated Food Photo Dataset

In the Japanese datasets, we collected about 83,000 calorie-annotated recipe data from six recipe sites (“Ajinomoto”†, “e-Recipe”††, “Kikkoman”†††, “Kyou no RR”††††, “Orange Page”†††††, “Lettuce Club”††††††). Each recipe presented in these sites contains an ingredient list, descriptions on cooking directions, food images, and the value of a food calorie as shown in Fig. 3. These websites do not provide the method for general users to post recipe information. All of the recipe sites except for “Ajinomoto” clearly indicate that professionals such as chefs provided recipe information which is expected to be reliable. Observing the collected data, it was found that most of the food photos contained one kind of dishes, and the value of food calorie corresponds to one serving. Therefore, in this study, we assume that a food photo has a single food label and we estimate a calorie value for one person. Note that the raw collected data includes some exceptions, and we removed them by hand.

We collect recipe data which has food calories information for one person. Then, we manually excluded the images with low resolution or multiple kinds of dishes. Finally, we excluded the food categories the number of samples of which was less than 100. In the end, a total of 4877 images were collected on 15 categories as shown in Fig. 4. Figure 5 (a) shows the calorie distribution of all the collected recipes, and Fig. 5 (b) shows that of “Miso soup”, “Spaghetti” and “Curry”. The calories of “Miso soup” are less than 200 kcal, while the calories of other foods than “Miso soup” are distributed around 500 kcal with the shape of the normal distribution. As shown in Fig. 5 (b), the distributions of values of food calories heavily depend on food categories. The distribution of some categories such as “Curry” are relatively broad.

4.2 American Calorie-Annotated Food Photo Dataset

In the American datasets, about 24,000 calorie-annotated recipe data were collected from Allrecipes∗∗∗∗. Allrecipes is a recipe site of a user contribution style, and the values of food calorie per serving is obtained from each recipe.
We used the categories used in Allrecipes and excluded the images with low resolution or multiple-dishes. In the end, a total of 2484 images were collected on 21 categories as shown in Fig. 6. Compared to the Japanese recipe datasets, this datasets contains foods which are more visually similar to each other.

5. Experiments

In this paper, we extended VGG-16 [19] and implemented multi-task CNN as shown in Fig. 2. After fc6 layers and fc7 layers, we inserted Batch Normalization layer [20] instead of using Dropout [21] in training time. In the layers other than Batch Normalization layers and the fc8 layers, the pre-trained weights for the ImageNet1000 classification task were used as initial values. For optimization of the CNNs, we used SGD with the momentum value, 0.9 and the size of mini-batch was 8. The weights of the loss term of Eq. (1) and Eq. (2) were determined as follows. Firstly, the weights of the loss terms are set to 1 and train once. In the training, the values of the losses for each iteration are preserved. Finally, the inverse of the average value of the loss in all iterations is used as the weight for the loss term of each task. In this experiments, we fixed \( \lambda \) to 1.

For the test, 10 models obtained at the 100 iteration intervals from the last 1k iterations in training were used, and the average value of the estimated values obtained from each model was taken as the final estimated value.

For the experiments on the proposed methods and baselines, we used Chainer\(^1\) [22] as a deep learning framework.

5.1 The Loss Function of Food Calorie Estimation

In this experiment, we tested the effectiveness of loss function combining \( L_{\text{rec}} \) and \( L_{\text{ab}} \). Equation (2) are compared to the loss function composed of each loss function. We used the Japanese recipe dataset described in Sect. 4.1. We used 70\% of the dataset for training, and the rest for evaluation. We used 0.001 of the learning rate for 50k iterations, and then used 0.0001 for 20k iterations.

Table 1 shows the result of food calorie estimation. We show the average of the relative error representing the ratio between the estimated values and the ground-truth, and the absolute error representing the differences between both. In addition we show the correlation coefficient between estimated value and ground-truth and the ratio of the estimated value within the relative error of 20\% and 40\%. In Table 1, the absolute error and the ratio of the estimated value within the relative error of 20\% indicate the accuracy is improved by using both errors. Therefore, in this paper, we use Eq. (2) as the loss function of food calorie estimation.

5.2 Cooking Direction Vector

In this experiment, we tested the effective cooking directions vector for food calorie estimation. We used the Japanese recipe dataset. We used 70\% of the dataset for training, and the rest for performance evaluation. We used 0.001 of the learning rate for 50k iterations, and then used 0.0001 for 20k iterations. we trained Word2Vec with about 8,710,000 sentences of cooking directions in CookPad recipe dataset.

The dimension of the word vector is \( n = 500 \). The average value of the number of words in sentences of cooking directions extracted from one recipe data was 44, we used the words of the top 44 regarding the TD-IDF values in each recipe. Then, in order to take time information into account simply, the sentence is divided into \( m \) in time order, and for each divided sentence, created a cooking directions vector by Eq. (6). Finally, the divided vectors are concatenated.

Table 2 shows the result of food calorie estimation with different values of \( m \). The table shows the average of the relative error and the absolute error, the correlation coefficient and the ratio of the estimated value within the relative error of 20\% and 40\%. Since the sentences of cooking directions are basically short, the positive effect of temporal division of cooking direction vectors was not obtained. Therefore, in this paper, we use \( m = 1 \) for constructing cooking direction vectors.

\(^1\)http://chainer.org/

---

![Fig. 6 21 categories of American recipe datasets.](image-url)
Table 3 The estimation results in Japanese recipe dataset.

|               | rel. err. (%) | abs. err. (kcal) | correlation | Top-1(%) |
|---------------|---------------|------------------|-------------|----------|
| calorie(single)| 29.4          | 100.7            | 0.778       | —        |
| +categories   | 27.9          | 95.2             | 0.802       | 45.9     |
| ++ingredients | 27.6          | 94.4             | 0.811       | 49.5     |
| +++directions | 27.4          | **91.2**         | **0.817**   | **50.1** |
| +ingredients  | 29.2          | 96.8             | 0.795       | —        |
| +directions   | 28.0          | 97.9             | 0.806       | 47.2     |
| ++directions  | 28.2          | 95.5             | 0.808       | 48.1     |
| ++categories  | 27.3          | 96.0             | 0.808       | 84.8     |
| categories(same)| —             | —                | —           | 81.2     |

5.3 Food Calorie Estimation with Japanese Recipe Dataset

We used the Japanese recipe dataset 4.1. We used 70% of the dataset for training, and the rest 30% for performance evaluation. We used 0.001 of the learning rate for 50k iterations, and then used 0.0001 for 20k iterations. For food ingredient vector and cooking directions vector, we trained Word2Vec with about 8,710,000 sentences of cooking directions in CookPad recipe dataset. The dimension of the word vector is \( n = 500 \). Because the average number of ingredients of the recipes was 12, we used the words of the top 12 ingredient names regarding TF-IDF values for each recipe data, and created a food ingredient vector by Eq. (6). In the same way as the previous experiments, for sentences of cooking directions for each recipe, we used the words of the top 44 regarding TF-IDF values, and created cooking directions vectors.

Table 3 shows the results of food calorie estimation with various combinations of auxiliary information. Regarding the food calorie estimation, we use the average of the relative error and the absolute error, the correlation coefficient and the ratio of the estimated value within the relative error of 20% for evaluation. In addition to food category estimation, we show the top-1 classification accuracy of food categorization in the rightmost column of Table 3. It indicates that the classification performance was improved by multi-task CNNs in any combinations which contained category information. In case of the multi-task CNN with all the four information, 2.0% and 9.5 kcal were reduced on the relative error and the absolute error, and 0.039 and 4.2% were increased on the correlation coefficient and the ratio of the estimated calories within 20% error, respectively. In addition, 2.9% were increased on the top-1 accuracy. Figure 7 (a) shows the relation between the ground truth values and the estimated calorie values by the single-task CNN, while Fig. 7 (b) shows the relation by the multi-task CNN. Note that the ellipses in both figures show the 95% confidence ellipse. Compared between both figures, we can confirm that the accuracy is improved by introducing of multi-task CNN since the shape of the 95% confidence ellipse becomes narrower. Figure 8 and Fig. 9 shows four examples of successfully estimated results and wrongly estimated results by multitask CNN with all the information, respectively.

5.4 Food Calorie Estimation with American Recipe Dataset

In the next experiments, we used the American recipe dataset explained in Sect. 4.2. We used 5-fold cross-validation for this dataset, since the size of test data is relatively small. In each fold, we used 80% of all the sample for training, and the rest 20% for testing. We used 0.001 of the learning rate for 30k iterations, and then used 0.0001 for 40k iterations. For food ingredient vector and cooking directions vector, we trained Word2Vec with about 82,000 sentences of cooking directions in American recipe dataset. The dimension of the word vector was 500. Because the average number of ingredients in one recipe was 300. We used the words vectors.
of the top 26 ingredient names regarding the TF-IDF value in each recipe data. In the same way, we used the words of the top 66 words on the TF-IDF values for constructing of cooking direction vectors.

We performed 5-fold cross-validation on food calorie estimation. The results are shown in the Table 4. In case of multi-task of food categories and ingredients, 1.6% and 8.9 kcal were reduced on the relative error and the absolute error, and 0.090 and 2.9% were increased on the correlation coefficient and the ratio of the estimated calories within 20% error, respectively. In addition, 6.7% were increased on the top-1 accuracy. Similar to Japanese recipe dataset, the effectiveness of multi-task CNNs was confirmed.

5.5 Comparison with the Image-Search-Based Method

In this experiments, we compared with the image-search based calorie estimation proposed by Miyazaki et al. [9]. They estimated food calories from food photos directly without estimating food categories and volumes. They searched the calorie-annotated photo database based on conventional hand-crafted features such as SURF-based BoF and color histograms, and estimated food calories by averaging the food calories of the top k similar food photos. Since they have not released the dataset used in [9], we re-implemented their method with state-of-the-art CNN features instead of hand-crafted features, and applied it to the Japanese recipe dataset for fair comparison.

In the experiments, we use VGG16 [19] which is pre-trained with the ImageNet 1000-class dataset for a feature extractor. Then, we extract activation signals of fully connected layers (fc layers) of the VGG16 network as CNN features. Both fc6 layer and fc7 layer of VGG16 [19] are 4096-dim, so we obtain a 4096-dim feature vector for each food image. Initially, we use 70% of the Japanese recipe dataset 4.1, to create a database of CNN features. Then, for each test image of the rest 30%, we gain the top k similar images by searching the database based on CNN features. Finally, we obtain a food calorie by calculating an average value of food calories of the top k similar images.

Table 5 shows the results. Our multi-task CNN outperformed the results by image-search based calorie estimation originally proposed by Miyazaki et al. [9]. About 20% and 19 kcal were reduced on the relative error and the absolute error, respectively.

6. Conclusions

In this paper, we proposed estimating food calorie from a food photo by simultaneous learning of food calories, categories, ingredients and cooking directions using multi-task CNNs. For experiments, we constructed two kinds of datasets that is a dataset of calorie-annotated recipe collected from Japanese recipe sites on the Web and a dataset collected from an American recipe site. In the experiments, in both datasets, multi-task CNNs outperformed independent single-task CNNs and the image-search based calorie estimation proposed by Miyazaki et al. [9].

We expect that much more accurate calorie estimation is possible by integrating multi-task CNN-based calorie estimation with volume/size-based calorie estimation approaches. As future work, we plan to introduce segmentation or multiple-view-based volume estimation into our framework. We also plan to implement a mobile application for real-time food calorie estimation by combining food dish detection.

References

[1] A. Myers, N. Johnston, V. Rathod, A. Kroattikara, A. Gorban, N. Silberman, S. Guadarrama, G. Papandreou, J. Huang, and P.K. Murphy, “Im2calories: towards an automated mobile vision food diary,” Proc. IEEE International Conference on Computer Vision, pp.1233–1241, 2015.

[2] K. Okamoto and K. Yanai, “An automatic calorie estimation system of food images on a smartphone,” Proc. ACM MM Workshop on Multimedia Assisted Dietary Management, pp.63–70, 2016.

[3] R. Tanno, K. Okamoto, and K. Yanai, “Deepfoodcam: A dcnn-based real-time mobile food recognition system,” Proc. ACM MM Workshop on Multimedia Assisted Dietary Management, p.89, 2016.

[4] J. Dehais, M. Anthimopoulos, and S. Mougiakakou, “Gocarb: A smartphone application for automatic assessment of carbohydrate intake,” Proc. ACM MM Workshop on Multimedia Assisted Dietary Management, p.91 2016.

[5] F. Kong and J. Tan, “Dietcam: Automatic dietary assessment with mobile camera phones,” Proc. Pervasive and Mobile Computing, pp.147–163, 2012.

Table 4 The estimation results in American recipe dataset.

|                 | rel. err. (%) | abs. err. (kcal) | correlation | 20% err. (%) | Top-1 (%) |
|-----------------|--------------|-----------------|-------------|--------------|-----------|
| calorie(single) | 43.3         | 128.5           | 0.293       | 32.2         | —         |
| +categories     | 42.9         | 120.5           | 0.361       | 34.3         | 58.7      |
| ++ingredients   | 41.7         | 119.6           | 0.383       | 35.1         | 61.1      |
| +++directions   | 42.9         | 120.6           | 0.369       | 33.7         | 61.3      |
| +ingredients    | 43.0         | 124.2           | 0.335       | 32.5         | —         |
| +directions     | 42.1         | 122.5           | 0.351       | 32.5         | —         |
| ++categories    | 42.1         | 123.0           | 0.349       | 33.6         | —         |
| categories(sing  | 42.4         | 120.8           | 0.365       | 33.5         | 59.3      |
|                | —            | —               | —           | —            | 54.4      |

Table 5 Comparison with a baseline of an image-search based calorie estimation.

|               | rel. err. (%) | abs. err. (kcal) | correlation | 20% err. (%) | Top-1 (%) |
|---------------|--------------|-----------------|-------------|--------------|-----------|
| VGG16(ImageNet) fc6 k=5 | 47.9         | 117.2           | 0.673       | 43.1         |
| VGG16(ImageNet) fc6 k=10 | 47.4         | 111.9           | 0.699       | 45.4         |
| VGG16(ImageNet) fc6 k=15 | 47.4         | 110.4           | 0.707       | 45.6         |
| VGG16(ImageNet) fc7 k=5  | 52.5         | 119.2           | 0.657       | 42.8         |
| VGG16(ImageNet) fc7 k=10 | 52.3         | 116.5           | 0.675       | 44.2         |
| VGG16(ImageNet) fc7 k=15 | 54.0         | 116.3           | 0.672       | 44.0         |
| Single-task CNN  | 29.4         | 100.7           | 0.778       | 45.9         |
| Multi-task CNN   | 27.4         | 91.2            | 0.817       | 50.1         |
[6] V. Bettadapura, E. Thomaz, A. Parnami, G.D. Abowd, and I. Essa, “Leveraging context to support automated food recognition in restaurant,” Proc. 2015 IEEE Winter Conference on Applications of Computer Vision (WACV), pp.580–587, 2015.

[7] M.-Y. Chen, Y.-H. Yang, C.-J. Ho, S.-H. Wang, S.M. Liu, E. Chang, C.-H. Yeh, and M. Ouhyoung, “Automatic chinese food identification and quantity estimation,” Proc. SIGGRAPH Asia Technical Briefs, pp.1–4, 2012.

[8] P. Poulaeddad, S. Shirmohammadi, and R. Al-maghrabi, “Measuring calorie and nutrition from food image,” IEEE Transactions on Instrumentation and Measurement, vol.63, no.8, pp.1947–1956, 2014.

[9] T. Miyazaki, G.C de Silva, and K. Aizawa, “Image-based calorie content estimation for dietary assessment,” Proc. IEEE ISM Workshop on Multimedia for Cooking and Eating Activities, pp.363–368, 2011.

[10] A.H. Abdulnabi, G. Wang, J. Lu, and K. Jia, “Multi-task CNN model for attribute prediction,” IEEE Transactions on Multimedia, vol.17, no.11, pp.1949–1959, 2015.

[11] J. Chen and C.-W. Ngo, “Deep-based ingredient recognition for cooking recipe retrieval,” Proc. ACM International Conference Multimedia, pp.32–41, 2016.

[12] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” Advances in Neural Information Processing Systems, 2013.

[13] L. Bossard, M. Guillaumin, and L. Van Gool, “Food-101 – mining discriminative components with random forests,” Proc. European Conference on Computer Vision, vol.8694, pp.446–461, 2014.

[14] Y. Matsuda, H. Hajime, and K. Yanai, “Recognition of multiple food images by detecting candidate regions,” Proc. IEEE International Conference on Multimedia and Expo, pp.25–30, 2012.

[15] T. Ege and K. Yanai, “Simultaneous estimation of food categories and calories with multi-task cnn,” Proc. IAPR International Conference on Machine Vision Applications (MVA), pp.198–201, 2017.

[16] T. Ege and K. Yanai, “Image-based food calorie estimation using knowledge on food categories, ingredients and cooking directions,” Proc. ACM Multimedia Thematic Workshops on Understanding, pp.367–375, 2017.

[17] L.C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A.L. Yuille, “Semantic image segmentation with deep convolutional nets and fully connected CRFs,” Proc. International Conference on Learning Representation, 2014.

[18] D. Eigen and R. Fergus, “Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture,” Proc. IEEE International Conference on Computer Vision, pp.2650–2658, 2015.

[19] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556, 2014.

[20] S. Ioffe and C. Szegedy, “Batch Normalization: Accelerating deep network training by reducing internal covariate shift,” Proc. International Conference on Machine Learning, 2015.

[21] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” Journal of Machine Learning Research, vol.15, pp.1929–1958, 2014.

[22] S. Tokui, K. Oono, S. Hido, and J. Clayton, “Chainer: a next-generation open source framework for deep learning,” Proc. NIPS Workshop on Machine Learning Systems, 2015.

Takumi Ege is a master course student at Department of Informatics, the University of Electro-Communications, Tokyo, Japan. He is working on food image recognition.

Keiji Yanai is a professor at Department of Informatics, the University of Electro-Communications, Tokyo, Japan. He 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 and till 2015 he was an associate professor at Department of Computer Science, the University of Electro-Communications, Tokyo. From November, 2003 to September, 2004, he was a visiting scholar at Department of Computer Science, University of Arizona, USA. His recent research interests include object recognition, deep learning, and Web multimedia mining.