IEICE TRANS. INF. & SYST., VOL.E102–D, NO.7 JULY 2019

PAPER Special Section on Multimedia for Cooking and Eating Activities

Webly-Supervised Food Detection with Foodness Proposal

Wataru SHIMODA[†a)], *Nonmember and* **Keiji YANAI**[†b)], *Member*

SUMMARY To minimize the annotation costs associated with training semantic segmentation models and object detection models, weakly supervised detection and weakly supervised segmentation approaches have been extensively studied. However most of these approaches assume that the domain between training and testing is the same, which at times results in considerable performance drops. For example, if we train an object detection network using only web images showing a large object at the center, it can be difficult for the network to detect multiple small objects. In this paper, we focus on training a CNN with only web images and achieve object detection in the wild. A proposal-based approach can address the problem associated with differences in domains because web images are similar to images of the proposal. In both domains, the target object is located at the center of the image and the ratio of the size of the target object to the size of the image is large. Several proposal methods have been proposed to detect regions with high "object-ness." However, many of these proposals generate a large number of candidates to increase the recall rate. Considering the recent advent of deep CNNs, methods that generate a large number of proposals exhibit problems in terms of processing time for practical use. Therefore, we propose a CNN-based "food-ness" proposal method in this paper that requires neither pixel-wise annotation nor bounding box annotation. Our method generates proposals through backpropagation and most of these proposals focus only on food objects. In addition, we can easily control the number of proposals. Through experiments, we trained a network model using only web images and tested the model on the UEC FOOD 100 dataset. We demonstrate that the proposed method achieves high performance compared to traditional proposal methods in terms of the trade-off between accuracy and computational cost. Therefore, in this paper, we propose an intermediate approach between the traditional proposal approach and the fully convolutional approach. In particular, we propose a novel proposal method that generates high"food-ness" regions using fully convolutional networks based on the backward approach by training food images gathered from the web.

key words: *food segmentation, convolutional neural network, deep learning, UEC-FOOD100*

1. Introduction

Recording daily eating habits using smart devices has recently become common practice. Food records can provide numerical data such as the number of calories consumed and the nutritional value of the food consumed. Such numerical data are useful in nutrition analysis and promotes healthy-eating habits. While food recordings can be useful, the process of recording can be laborious. It is unrealistic to expect that everyone indicates all the dish names they consume in their daily meals through texts. Thus, we require a smarter approach to keep record of daily meals.

Food image recognition plays an important role in the simplification of food recordings. If we can replace the manual procedure of taking a picture, we can dramatically reduce the burden associated with food recording, freeing us from the laborious procedure even though it is needed for every meal consumed. To simplify food recording also matches the recent fashion so that there are trends uploading food images to SNS. In terms of technical aspects, food image recognition also matches the recent trends in fashion, owing to recent significant advances in deep neural networks. Deep convolutional neural networks (DCNN) have been used for large-scale object recognition at the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) 2012. Krizhevsky et al. [1] were the winners of ILSVRC 2012, outperforming other teams who employed conventional hand-crafted feature approaches. Considerable advances in DCNN approaches still continue since that success. In the ILSVRC Challenge 2015, Szegedy et al. [2] and Simonyan et al. [3] overcame the human score. In the ILSVRC Challenge 2016, He et al. [4] achieved state-of-the-art performance, including human performance with over one hundred layers. Moreover, DCNN has also outperformed the state-of-the-art approaches in other tasks associated with computer vision. In particular, object detection and semantic segmentation tasks have been studied using applied methods based on DCNN, achieving significant progress.

In food recognition, object detection and semantic segmentation are also important tasks. The detection task involves a bounding box with a target object, and the semantic segmentation task predicts each pixel belonging to a particular class. We can estimate the food position and obtain food sizes at a bounding-box-level or pixel-level using the results of these tasks. In particular, we consider the prediction of food sizes to be important for food recognition in the sense that food sizes must be related to the food amount. Precise food calorie estimation is a promising field in food recording. Knowledge of food amount in terms of calories has been widely accepted as common understanding. Therefore, object detection and semantic segmentation will result in enhanced estimation of calories.

However, most CNN-based object detection and semantic segmentation methods assume that additional annotation is available in the form of bounding-box annotation and pixel-wise annotation, which can be costly to obtain in

Manuscript received October 12, 2018.
Manuscript revised January 22, 2019.
Manuscript publicized April 25, 2019.

[†]The authors are with Department of Informatics, The University of Electro-Communications, Chofu-shi, 182–8585 Japan.
a) E-mail: shimoda-k@mm.inf.uec.ac.jp
b) E-mail: yanai@cs.uec.ac.jp

DOI: 10.1587/transinf.2018CEP0001

general. On the other hand, collecting images with image-level annotation is relatively easier than pixel-level annotation because many images with attached tags are available on hand-crafted open image data sets such as ImageNet and on the web. In this study, we focus on weakly-supervised semantic segmentation, which requires neither pixel-wise annotation nor bounding box annotation but only image-level annotation.

In general, object detection and semantic segmentation with bounding-box annotation or pixel-wise annotation are referred to as fully supervised methods, while object detection and semantic segmentation with only image-level annotation are referred to as weakly supervised methods.

In recent years, some weakly supervised object detection and semantic segmentation methods with DCNN have been proposed. However, most of the previous works were tested on only the Pascal VOC 2012 dataset. Although the Pascal VOC 2012 dataset includes multi-labeled images, most Web images include only a single label. Therefore, some weakly supervised methods are not stable because they are trained using only Web images. Training methods that use only Web images are often referred to as Webly supervised methods [5]. In this paper, we focus on Webly supervised detection and segmentation.

In particular, we consider "Distinct Class-specific Saliency Maps (DCSM)" [6] to be weakly supervised detection and segmentation methods. Such methods demonstrate high performance in weakly supervised tasks and can be easily used to adapt to other targets. However DCSM is ineffective for Webly supervised methods because of the change in domain at the time of training and testing. In Webly supervised approaches, most training images are single labeled images and we assume that the targets are multiple-food images, consisting of multiple foods in the test phase. The differences in the domain can cause considerable performance drops. However, we determined that we can obtain the rough food regions from the outputs of the DCSM, even though it is difficult to directly obtain detailed food regions and the correct class of food for the region. We consider the rough food regions to be a type of proposal for food objects and we define "food-ness" as a representation that reflects how likely a pixel belongs to a region of any food category. In this paper, we used "food-ness" as a proposal for foods and we apply it to a proposal-based method for foods by following traditional detection or segmentation methods such as RCNN and SDS. For this proposal method, we primarily discuss the computational costs and the methods of generating a small number of effective region candidates. Note that this paper is based on our previous conference paper [7] with the revisions on related works and the explanation of the proposed method.

We summarize our contributions as below:

- We achieved Webly supervised food-detection and food-segmentation for the first time.
- We proposed a novel proposal method for food images.

2. Related Works

In this paper, we focus on image recognition in the domain of food. Our study is also related to object detection and semantic segmentation. In terms of related works, we discuss previous food recognition studies, including food detection and segmentation, and recent CNN-based detection and segmentation works for generic images.

2.1 Food Recognition

Food image recognition is a promising application of visual object recognition, owing to its potential in estimating food calories and analyzing the eating habits of people for their general well-being. There have been numerous studies on food image recognition that have been published [8]–[14].

Moreover, the effectiveness of deep convolutional neural networks (DCNN) has been recently demonstrated for large-scale object recognition at the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) 2012. Krizhevsky et al. [1] won the ILSVRC 2012, outperforming all other teams who employed conventional hand-crafted feature approaches. In the DCNN approach, input data consist of a resized image, and the output is a class-label probability. In other words, DCNN includes all the object recognition steps such as local feature extraction, feature coding, and learning. In general, the advantages of DCNN includes the adaptive estimation of optimal feature representations for datasets, which is not possible using conventional hand-crafted feature approaches. In conventional approaches, we first extract local features such as SIFT and SURF and then code them into bag-of-feature or Fisher Vector representations. In the context of food image recognition, classification accuracy based on the UEC-FOOD100 dataset [11] improved from 59.6% [10] to 72.26% [15] by replacing the Fisher Vector and linear SVM with DCNN.

However, most studies assume that one food image represents only one food item. The approaches presented in these studies cannot handle an image that contains two or more food items such as an image of a hamburger and French fries. To list all food items in a given image of food and to estimate the calories associated with the food, the segmentation of food is needed. Some studies attempted food region segmentation [11], [16]–[18].

Matsuda et al. [11] proposed the use of multiple methods to detect food regions, including Felzenszwalb's deformable part model (DPM) [19], a circle detector, and the JSEG region segmentation method [20].

He et al. [18] employed local variation [21] to segment food regions for estimating the total calories associated with the food in a given food photo. In some studies on mobile food recognition [16], [17], users were asked to point to the rough locations of each food item in an image of food and to perform GrabCut [22] for extracting food item segments.

In addition, there have been several studies on the estimation of calories using computer vision techniques. Kong

et al. [23] reconstructed 3D food models using multi-angle pictures and estimated the calories associated with the food using the cubic volume of 3D models. Chen et al. [24] recognized an image and computed the cubic volume using depth information. It must be noted that they obtained depth information using a sensor. 3D base calorie estimation methods tend to be laborious for users. On the other hand, Myers et al. [25] proposed a calorie estimation application called "im2calorie." They obtained each pixel depth information through deep learning prediction and estimated the food calories. However Myers et al. have not achieved practical use.

Pouladzadeh et al. [26] estimated food calories from the segmentation results of an image. They defined a thumb as the base food area and estimated food volumes and calories from the area ratios of the thumb and the food. While we can always take a picture of food using our thumbs, this method can potentially distort the image and taking a picture with only one hand can be difficult. As more recent study, Myers et al. [25] proposed calorie estimation application which called "im2calorie". They obtained each pixel depth information by prediction of deep learning and estimated calories. In contrast to previous studies, we tackled food image segmentation with limited annotation through Webly supervised learning.

2.2 CNN-Based Fully-Supervised Object Detection and Semantic Segmentation

As early works on CNN-based semantic segmentation, Girshick et al. [27] and Hariharan et al. [28] proposed object segmentation methods using region proposal and CNN-based image classification. They first generated at most 2000 region candidates using selective search [29] and then applied CNN image classification through the feed-forwarding of the CNN to each of the proposals. They finally integrated all the classification results using non-maximum suppression and generated the final object regions. Although these methods significantly outperformed the conventional methods, they had a drawback in which they required long processing times for CNN-based image classification of many region proposals.

While Girshick et al. [27] and Hariharan et al. [28] took advantage of the excellent ability of a CNN for image classification tasks involved in semantic image segmentation in a relatively straightforward manner, He et al. [30] and Long et al. [31] proposed CNN-based semantic segmentation in a hierarchical manner. A CNN is much different from a conventional bag-of-features framework in terms of the multi-layered structure consisting of multiple convolutional and pooling layers. Because a CNN has several pooling layers, the location information is gradually lost as the signal is transmitted from the lower layers to the upper layers. In general, the lower layers hold location information in their activations, while the upper layers hold weak local information. He et al. proposed spatial pyramid pooling that exploits lower layer information for object detection and reduced large computational costs associated with an RCNN [27]. Long et al. [31] replaced the fully connected layers in the convolutional layers and directly learned the matrix outputs of the fully convolutional networks through pixel-wise-annotation, which is often referred to as end-to-end network. Later, Ren et al. [32] proposed faster RCNN, which is an end-to-end network for object detection tasks.

2.3 CNN-Based Weakly-Supervised Semantic Segmentation

Most conventional non-CNN-based weakly supervised segmentation methods employ a conditional random field (CRF) with unary potentials estimated through multiple instance learning [33], extremely randomized hashing forest [34], and GMM [35].

As a CNN-based method, Pedro et al. [36] achieved weakly-supervised segmentation by using multi-scale CNN proposed in [37]. They integrated the outputs that contain location information with log sum exponential and limited object regions to the regions overlapped with object proposals [38].

Pathak et al. [39], [40] and Papandreou et al. [41] achieved weakly-supervised semantic segmentation by adapting CNN models for fully-supervised segmentation to weakly-supervised segmentation. In MIL-FCN [39], they trained the CNN for full-supervised segmentation proposed in Long et al. [31] with a global max-pooling loss that enables the training of the CNN model using only training data with image-level labels. Constrained convolutional neural networks (CCNN) [40] improved MIL-FCN by adding constraints and using fully-connected CRF [42]. Papandreou et al. [41] trained the DeepLab model [43] proposed as a fully-supervised model with the EM algorithm, which is referred to as "EM-adopt." Both CCNN and EM-adopt generate pseudo-pixel-level labels from image-level labels using constraints and the EM algorithms to train FCN and DeepLab, which were originally proposed for fully supervised segmentation, respectively. Both demonstrated that dense CRF [42] are helpful in boosting segmentation performance even in a weakly supervised setting.

Meanwhile, Simonyan et al. [3] proposed a method of generating object saliency maps by back propagation (BP) over a pre-trained DCNN and demonstrated semantic object segmentation by applying GrabCut [22] using saliency maps as seeds. While all the above-mentioned methods of weakly supervised segmentation employ only feed-forward computation, Shimoda et al. [6] adopted a method based on back-propagation (BP) computation and was successful.

Although the above-mentioned weakly supervised methods achieved remarkable progress, their performance was tested only on the Pascal VOC 2012 dataset, i.e. using multi-label training images including only general object class. Therefore we propose a Webly supervised food object detection method, which requires only web images for training. We consider the combination of traditional proposal-based approaches and fully convolutional approaches. We

demonstrated that our method is robust to changes in domains. We trained the parameters of the CNN with only single-labeled web images and performed tests using multi-labeled images.

2.4 CNN-Based Weakly-Supervised Object Detection

Several weakly supervised object detection methods have been recently proposed. Bilen et al. proposed WSDDN [44], which is a method of training an object detection network under image level supervision. They trained classification data streams and detection data streams using a novel loss. Diba et al. [45] extended WSDDN [44]. They used weakly supervised segmentation for improving the accuracy of weakly supervised detection. The proposed method is divided into an end-to-end 3-stage cascaded CNN and the weakly supervised segmentation factor is used in the second stage for refinement of the bounding box. Kim et al. [46] also improved the object detection accuracy by using weakly supervised segmentation. They separated training into two phases, which was a different approach from previous weakly supervised detection methods. Kantorov et al. [47] proposed an architecture that used the surrounding contexts of ROI. Although their architecture is simple, its accuracy has been greatly improved. Previous weakly supervised methods focus on directly training object detection networks. However this approach exhibits limitations such as differences between the domain of training images and test images and these differences frequently result in a significant deterioration of detection accuracy. For example, if we train an object detection network with only web images that have a large object at the center, it can be challenging for the network to detect multiple small objects. Different from these weakly supervised object detection methods, our method is robust to differences in domain changes to focus on learning the concept of "food-ness."

3. Proposed Method

We propose a new method of generating "food-ness" regions with weakly supervised annotation. Our method is based on distinct class-specific saliency maps (DCSM) [6], which is an extension of Simonyan et al. [3]. In this section, we discuss the DCSM and the manner in which DCSM has been adopted for "food-ness" proposal.

3.1 Overall Architecture

We follow traditional detection methods by using proposals. We first generate proposals based on DCSM. We then identify each candidate region. Finally, we unify overlapped candidates by Non Maximum Suppression (NMS). In this study, we prepare two CNNs for proposal and recognition. We illustrate an overview in Fig. 1. Details of the proposed method process is as follows:

- Recognize an image.

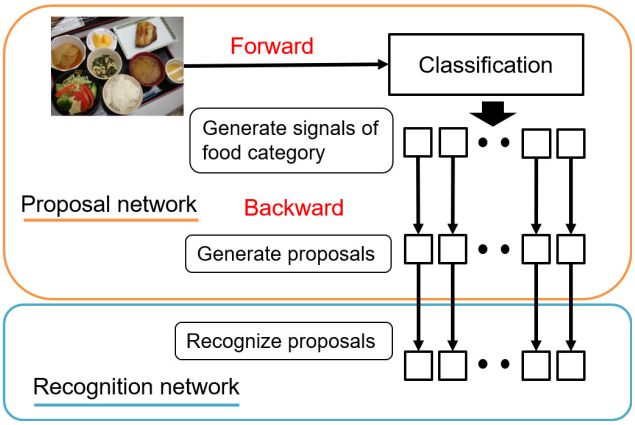

Fig. 1 Processing flow of our method.

- Sort each food class based on the softmax output.
- Back-propagate upper rank class score.
- Subtract each class derivative value.
- Obtain "food-ness" proposals.
- Recognize each "food-ness" candidate.
- Unify overlapped candidates by NMS.

3.2 DCSM

In [3], the authors considered the derivatives of the class score with respect to the input image as class saliency maps. However, the position of an input image is the furthest from the class score output on the deep CNN, which sometimes causes weakening or vanishing of gradients. Instead of the derivatives of the class score with respect to the input image, Shimoda et al. [6] used the derivatives with respect to feature maps of the relatively upper intermediate layers that are expected to retain more high-level semantic information. In addition, they applied some techniques that are known to be effective in semantic segmentation through a backward approach. They selected the maximum absolute values of the derivatives with respect to the feature maps at each location of the feature maps across all the kernels and up-sampled them with bi-linear interpolation so that their size becomes the same as an input image.

The class score derivative v_i^c of a feature map layer is the derivative of the class score Sc with respect to the layerL_i at the point (activation signal) L_i^0:

$$v_i^c = \left.\frac{\partial S_c}{\partial L_i}\right|_{L_i^0} \tag{1}$$

v_i^c can be computed by back-propagation. After obtaining v_i^c, Shimoda et al. up-sampled it to w_i^c through bi-linear interpolation so that the size of a 2-D map of v_i^c becomes the same as an input image. Next, the class saliency map $M_i^c \in \mathcal{R}^{m \times n}$ is computed as

$$M_{i,x,y}^c = \max_{k_i} |w_{i,h_i(x,y,k)}^c|, \tag{2}$$

where $h_i(x, y, k)$ is the index of the element of w_i^c.

The saliency maps of two or more different classes tend to be similar, particularly at the image level. The saliency maps by [3] are likely to correspond to foreground regions rather than specific class regions. To address this, Shimoda et al. [6] proposed to subtracte saliency maps of the other candidate classes from the saliency maps of the target class to different target objects from other objects. They selected several candidate classes with a pre-defined threshold and a pre-defined minimum number.

The improved class saliency maps with respect to class c, $\tilde{M}_i^c$, are represented as:

$$\tilde{M}^c_{i,x,y} = \sum_{c' \in candidates} \max\left(M^c_{i,x,y} - M^{c'}_{i,x,y}, 0\right)[c \neq c'], \quad (3)$$

where *candidates* is a set of selected candidate classes. Subtraction of saliency maps resolved the overlapped regions among the maps of the different classes.

Shimoda et al. [6] used fully convolutional networks (FCN) that accept arbitrary-sized inputs for multi-scale generation of class saliency maps. If an input image that is larger than the one used in the original CNN is given to the fully-convolutional CNN, class score maps represented as $h \times w \times C$ are outputted, where C is the number of classes, and h and w are larger than 1.

To obtain CNN derivatives with respect to enlarged feature maps, Shimoda et al. [6] simply back-propagated the target class score map defined as $S_c(:,:,c) = 1$ (in the MATLAB notation) with 0 for all other elements, where c is the target class index.

The final class saliency map $\hat{M}^c$ averaged over the layers and the scales is obtained as follows:

$$\hat{M}^c_{x,y} = \frac{1}{|S||L|} \sum_{j \in S} \sum_{i \in L} \tanh(\alpha \tilde{M}^c_{j,i,x,y}), \quad (4)$$

where L is a set of the layers for which saliency maps are extracted, S is a set of the scale ratios, and α is a constant which we set to 3 in the experiments. Note that we assume the size of $\tilde{M}_{j,i}$ for all the layers are normalized to the same size as an input image before calculation of Eq. (4).

In [48], guided back-propagation (GBP) [48] was adopted as a back-propagation method instead of normal back-propagation (BP) used in [3]. The difference between the two methods is the backward computation through ReLU. GBP can visualize saliency maps with fewer noise components than normal BP by back-propagating only the positive values of CNN derivatives through ReLU [48].

3.3 "Food-ness" Proposal

In this paper, we focus on training models with single-food images and on testing multiple-foods images. In general, domain changes from training time to testing time results in performance degradation. This problem is referred to as one of the cross-domain problems or the domain adaptation problems. Using the DCSM, this problem was also observed and accuracy degraded significantly. We illustrate our situation using this domain adaptation problem and an example at Fig. 2 in food images.

Fig. 2 Example of our cross domain situation.

In this study, we avoid this domain adaptation problem using region proposals. Proposal methods generate object region candidates and these candidates must include target objects. When recognizing target objects in the candidates, we obtain better results than in the case of recognizing raw images without proposals. Because, in our situation, test images include multiple food images, some candidate regions can be considered single food images. Therefore, the condition with some candidate regions is closer to the training condition than the raw test images condition.

RCNN [49] and SDS [28] are typical methods of detection and segmentation using proposals based on DCNN. They use selective search [29] and MCG [38] as proposal methods. These proposal methods also typically generate a considerable number of candidates, approximately 2000 with local features. A considerable number of candidates rise recall but pay off computational costs. We consider the number of candidates, approximately 2000, to be too large and there can be several inefficient processes for food recognition. Therefore, we propose a novel proposal method for foods with DCNN.

"Objectness" is a value that reflects the likelihood that a region or bounding box in an image covers an object of any category. In this study, we define "food-ness" as a representation that reflects the likelihood that a pixel belongs to a region of any food category. In this study, we adapt DCSM for calculating "food-ness."

The original DCSM approach is ineffective because of the problem of domain changes as we mentioned above. In fact, the estimated regions by DCSM trained with only Web images are not precise. However, we observed that most regions belonged to any food items in an image. Interestingly, the estimated regions for food classes that are not included in a given image still belong to other existing objects, and some regions fit food regions as shown in Fig. 3. This means that CNN trained with different domain images could not precisely transfer knowledge related to the category of food but could learn rough food conception.

In practice, to adapt DCSM for "food-ness" we increase the number of *candidates* in Eq. (3). It must be noted that we do not aggregate multi-input-scale results because

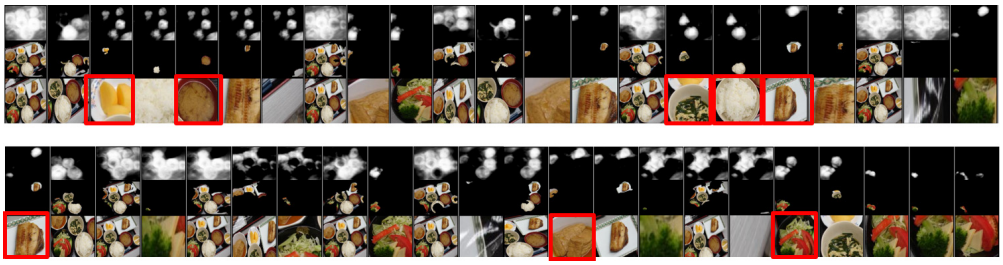

Fig. 3 Proposal results. The first row presents the saliency obtained from DCSM. The second row indicates the regions obtained from saliency maps. The third row indicates the bounding boxes that we recognize. The red rectangle indicates a good candidate.

of increasing computational costs. We obtain the probability maps for each signal of a class using backpropagation as follows:

$$P^c = \frac{1}{|L|}\sum_{i\in L}\tanh(\alpha M_i^c), \tag{5}$$

where P^c denotes probability maps such as saliency maps. We convert the probability maps P^c to masks M^c through thresholding. In this study, we set the threshold to 0.5. When the mask M^c contains multiple food items, the probability maps often include several peaks. Therefore, to obtain better proposals, we divide each mask M^c into several masks M_k^c by separating the isolated regions using a binary tracing method. $k \in \{1, 2, \cdots, K\}$ represents the elements of regions and K is the number of regions. For binary tracing, we used *bwconncomp*, which is a MATLAB function. We finally integrated the masks by ignoring the category of signals for backpropagation and we used the integrated masks $\hat{M}_{k'}$, $(K' = C \times K)$ as food-ness proposal.

To summarize this, we increase the number of candidate classes in the DCSM method and obtain regions from the output probabilities with DCSM. We can increase the number of candidate classes as far as the maximum target class number, which in our case is 100 for the UEC-FOOD100 dataset. We will discuss the manner in which the number of classes in Sect. 4.2 are chosen. For each input image $x \in \mathcal{X}$, we compute the proposals $\hat{M}_{k'}$ using the above process. We then obtain bounding boxes $\hat{B}_{k'}$ by extracting the maximum and minimum values of the coordinate from the pixels, which belong to the food region on each mask $\hat{M}_{k'}$. For each bounding box $\hat{B}_{k'}$, we cropped the images $x_{k'}^p$ and identified these cropped images using recognition networks. The training of recognition networks is independent of the proposal network. We train the proposal network and recognition network separately. Details of the training are presented in Sect. 4.

4. CNN Training

In this study, we adopt VGG16 as a base convolutional network for fine-tuning food images. Although there are common factors, we separate the proposal network and recognition network because of differences in applications. We fine-tune VGG16 as a proposal network with a fully convolutional technique. We also fine-tune VGG16 for recognition networks in a traditional way. In this section, we present the details concerning these two networks.

4.1 Proposal Network

As an off-the-shelf basic CNN architecture, we use the VGG-16 [50] pre-trained with 1000-class ILSVRC datasets. In our framework, we fine-tune a CNN with training images with only image-level annotation. Fully convolutional networks (FCN) that accept arbitrary-sized inputs have been recently used in studies on CNN-based detection and segmentation such as [51] and [31]. The fully connected layers in these studies with n units were replaced with equivalent convolutional layers having n 1×1 filters. Following these studies, we introduce FCN for multi-scale generation of class saliency maps. When training, we insert global max pooling before the final loss function layer to handle input images that are larger than the images used for pre-training of the VGG-16. Global max pooling is an operation that has been adopted in several weakly supervised segmentation methods. The purpose of this operation is to convert the last output to a vector from a matrix. Therefore, we can train FCN with usual image-level-label and soft-max loss.

In particular, we replace a fully connected layer with a convolution layer for the VGG16-model and train the network on the UECFOOD-100 dataset, which consists of 100 types of food classes with global max pooling.

4.2 Recognition Network

For recognition, although we change only the last layer for food category outputs, we prepare additional categories for training. The purpose of a recognition network is to discriminate candidates obtained from the proposal network. The conditions for recognizing candidates vary from the training phase in terms of including non-target-category-object images and small-food-patch images. In RCNN and SDS, they consider only non-target-category-object images as the background so that RCNN and SDS can be tested on a general object detection dataset. However, food recognition is different for general object recognition. Food recognition has the similarity to the texture recognition, namely, food patches can be discriminated as food with a high score by

Table 1 Mean average precision over all the 100 categories, 53 categories (more than 10 items of which are included in the test data), and 11 categories (more than 50 items of which are included in the test data) for the results in the different conditions and models.

method	small-patch class	low-resolution images	training with only web images	100class (all)	53class (#item ≥ 10)	11class (#item ≥ 50)
"Foodness 1"	-	-	-	30.0	29.3	31.9
"Foodness 2"	✓	-	-	33.7	39.0	33.6
"Foodness 3"	✓	✓	-	39.5	46.0	38.9
"Foodness 4"	-	-	✓	33.5	35.1	33.3
"Foodness 5"	✓	-	✓	32.2	34.8	31.8
"Foodness 6"	✓	✓	✓	36.4	39.9	36.3

Table 2 Comparison of global pooling operations for "food-ness".

method	training with only web images	100class (all)	53class (#item ≥ 10)	11class (#item ≥ 50)
"Foodness" (average pooling)	-	39.5	46.0	38.9
"Foodness" (average pooling)	✓	36.4	39.9	36.3
"Foodness" (max pooling)	-	39.9	48.3	37.6
"Foodness" (max pooling)	✓	38.9	42.5	38.1

DCNN. For example, in the case of dog recognition with DCNN, the recognition results for the proposals of legs and skin will include low scores in dog probability, while, in the case of food recognition, the patches of rice images will indicate high scores in rice probability. To sum up, DCNN cannot discriminate general objects with limited parts but can discriminate foods with minimum patch information. Therefore we create additional classes for food patch class.

Furthermore we add low-resolution images because we determined that a low-resolution image was discriminated as food patch category. We assume that this is the reason for which a small-food-patch image tends to be a low-resolution image. Therefore, we add low-resolution images to each food class. Our intuition is that if we consider low-resolution images to be training images, the low-resolution images will not be recognized as small-food-patch images.

We augment the training images for cropped images and expand the category to 202 from 101 to address some problems for each candidate recognition in food images. Practically, we cropped three images from each training image as a food path using random positions with random sizes. The minimum size of the cropped image is 50 and the maximum size is 150. It must be noted that the original image size is 256, i.e. the rate of each cropped image size for the original image is approximately 0.2 and 0.6. We also prepare three images as low-resolution images by down-sampling and rescaling. We randomly defined down-sample sizes. The minimum downscaled size is 10 and the maximum size is 256, which is equal to the original image size. We finally obtain augmented training images that are seven times larger than the original training images.

5. Experiments

In the experiments, we used the UEC-FOOD100 dataset [11] and web food images. The UEC-FOOD100 dataset [11] consist of 100 class food categories and each category includes 100 images. It should be noted that each food item is an annotated bounding box. On the other hand, although the web food images have the same category as in the UEC-FOOD100 dataset, each category includes 1000 images without bounding box annotation. Most of these web food images are obtained from twitter streams and some images are obtained from the Bing API. We use multiple-food images from the UEC-FOOD100 dataset as a test dataset for object detection. All detection evaluations based on mean average precision are also considered for Pascal VOC detection evaluation.

5.1 Food Detection Evaluation

We prepared two datasets, one dataset consist of UECFOOD-100 and web Images. Another dataset consist of only web images.

5.1.1 Additional Classes for Recognition Network

We first evaluate three cases of recognition networks with two datasets using a fixed proposal network setting. Table 1 presents the average precision (AP) of the three models trained under different conditions with two training data."Foodness 2" demonstrated higher performance than "Foodness 1". This means that adding a small patch class is effective. On the other hand, "Foodness" 3 achieved better results than Foodness 2". We can observe that adding low-resolution images is also effective for the recognition network."Foodness 4", "Foodness 5" and "Foodness 6" are trained with only Web images. The AP of "Foodness 6" is higher than "Foodness 4" and "Foodness 5". In Webly supervised, additional classes are also effective."Foodness 6" exhibits a drop in AP compared with "Foodness 3"; while overcoming the AP of "Foodness 2". Based on the results above, we can state that additional classes are effective and Webly supervised learning possesses reasonable capabilities.

Table 3 Comparison with other traditional proposal method.

method	100class (all)	53class (#item ≥ 10)	11class (#item ≥ 50)	proposal speed[s]	Recognition speed for candidates[s]
Selective Search [29]	38.3	39.1	35.7	7.6	35.0
Multiscale Combinatorial Grouping [38]	33.9	43.7	33.4	2.5	35.0
"Foodness" with 10 candidate classes	33.1	33.0	33.2	0.5	1.1
"Foodness" with 20 candidate classes	36.5	40.1	37.7	1.0	2.6
"Foodness" with 30 candidate classes	38.9	42.5	38.1	1.4	3.8

Fig. 4 Examples of results. Left images are input images. Center images are detection results. Right images are ground truth images.

5.1.2 Global Pooling for Proposal Network

We then compare two general global-pooling operations, global average pooling, and global max pooling. Table 2 presents a comparison of final pooling operations for two datasets.

5.1.3 Comparison with Other Traditional Proposal Methods

Next, we compare the quality of our proposal method with that of other traditional proposal methods. We evaluate our methods in terms of mean AP and speed factors. We prepare two traditional proposal methods as baselines. Selective search (SS) [29] is a bounding-box proposal method and Multiscale Combinatorial Grouping (MCG) [38] involves a segmentation region proposal method. Both methods generate a large number candidates, approximately 2000. To assess our proposal quality, we changed the candidate class number. Small candidate class results in smaller computational costs so that the time of backward computation can be reduced. Table 3 presents the comparison results. It must be noted that recognition speed includes theoretical values computed from candidate numbers and the computational cost of an image. AP of "Foodness" with 30 candidate classes outperforms SS [29] and MCG [38] even though it has 40 times lesser number of candidates. In addition, even if we reduced the candidate class number, the mean AP is still held by 30%. This shows that our proposal exhibits sufficient quality for "food-ness" detection.

6. Conclusions

We proposed a CNN-based "food-ness" proposal method that requires no pixel-wise annotation even in the case of bounding box annotation. We focused on an intermediate approach involving traditional proposal approaches and fully convolutional approaches. In particular, we proposed a novel proposal method that generates "food-ness" regions through a fully convolutional network-based backward approach by training web food images. Therefore, we achieved a reduction in computational costs and ensured quality food detection.

In future studies, we aim to focus on Webly-supervised food segmentation in addition to detection because our "food-ness" proposal can also generate segmentation results.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 15H05915, 17J10261, 17H01745, 17H05972, 17H06026, and 17H06100.

References

[1] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "Imagenet classification with deep convolutional neural networks," Commun. ACM, vol.60, no.6, pp.84–90, 2017.
[2] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," Proc. of IEEE Computer Vision and Pattern Recognition, 2015.
[3] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," Proc. of International Conference on Learning Represenation Workshop Track, 2014.
[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Proc. of IEEE Computer Vision and Pattern Recognition, pp.770–778, 2016.
[5] X. Chen and A. Gupta, "Webly supervised learning of convolutional

networks," Proc. of IEEE International Conference on Computer Vision, pp.1431–1439, 2015.
[6] W. Shimoda and K. Yanai, "Distinct class saliency maps for weakly supervised semantic segmentation," Proc. of European Conference on Computer Vision, pp.218–234, 2016.
[7] W. Shimoda and K. Yanai, "Foodness proposal for webly-supervised food detection," Proc. of ACM MM Workshop on Multimedia Assisted Dietary Management, pp.13–21, 2016.
[8] L. Bossard, M. Guillaumin, and L.V. Gool, "Food-101 - mining discriminative components with random forests," Proc. of European Conference on Computer Vision, vol.8694, pp.446–461, 2014.
[9] H. Kagaya, K. Aizawa, and M. Ogawa, "Food detection and recognition using convolutional neural network," Proc. of ACM International Conference Multimedia, pp.1085–1088, 2014.
[10] Y. Kawano and K. Yanai, "Foodcam: A real-time food recognition system on a smartphone," Multimedia Tools and Applications, vol.74, no.14, pp.5263–5287, 2015.
[11] Y. Matsuda, H. Hoashi, and K. Yanai, "Recognition of multiple-food images by detecting candidate regions," Proc. of IEEE International Conference on Multimedia and Expo, pp.25–30, 2012.
[12] 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," SIGGRAPH Asia, 2012.
[13] M. Bosch, F. Zhu, N. Khanna, C.J. Boushey, and E.J. Delp, "Combining global and local features for food identification in dietary assessment," Proc. of IEEE International Conference on Image Processing, pp.1789–1792, 2011.
[14] S. Yang, M. Chen, D. Pomerleau, and R. Sukthankar, "Food recognition using statistics of pairwise local features," Proc. of IEEE Computer Vision and Pattern Recognition, pp.2249–2256, 2010.
[15] Y. Kawano and K. Yanai, "Food image recognition with deep convolutional features," Proc. of ACM UbiComp Workshop on Workshop on Smart Technology for Cooking and Eating Activities (CEA), pp.589–593, 2014.
[16] C. Morikawa, H. Sugiyama, and K. Aizawa, "Food region segmentation in meal images using touch points," Proc. of ACM MM WS on Multimedia for Cooking and Eating Activities (CEA), pp.7–12, 2012.
[17] Y. Kawano and K. Yanai, "Real-time mobile food recognition system," Proc. of IEEE CVPR International Workshop on Mobile Vision (IWMV), pp.1–7, 2013.
[18] Y. He, C. Xu, N. Khanna, C.J. Boushey, and E.J. Delp, "Food image analysis: Segmentation, identification and weight estimation," Proc. of IEEE International Conference on Multimedia and Expo, pp.1–6, 2013.
[19] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," IEEE Trans. Pattern Anal. Mach. Intell., vol.32, no.9, pp.1627–1645, 2010.
[20] Y. Deng and B.S. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," IEEE Trans. Pattern Anal. Mach. Intell., vol.23, no.8, pp.800–810, 2001.
[21] P.F. Felzenszwalb and D.P. Huttenlocher, "Image segmentation using local variation," Proc. of IEEE Computer Vision and Pattern Recognition, pp.98–104, 1998.
[22] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," ACM Transactions on Graphics (TOG), vol.23, no.3, pp.309–314, 2004.
[23] F. Kong and J. Tan, "Dietcam: Automatic dietary assessment with mobile camera phones," Proc. of Pervasive and Mobile Computing, vol.8, no.1, pp.147–163, 2012.
[24] 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," SIGGRAPH Asia Technical Briefs, p.29, 2012.
[25] A. Myers, N. Johnston, V. Rathod, A. Korattikara, A. Gorban, N. Silberman, S. Guadarrama, G. Papandreou, J. Huang, and K. Murphy, "Im2calories: Towards an automated mobile vision food diary," The IEEE International Conference on Computer Vision (ICCV), pp.1233–1241, 2015.
[26] P. Pouladzadeh, S. Shirmohammadi, and R. Al-Maghrabi, "Measuring calorie and nutrition from food image," IEEE Trans. Instrum. Meas., pp.1947–1956, 2014.
[27] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," Proc. of IEEE Computer Vision and Pattern Recognition, pp.580–587, 2014.
[28] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," Proc. of European Conference on Computer Vision, vol.8695, pp.297–312, 2014.
[29] J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, and A.W.M. Smeulders, "Selective search for object recognition," International Journal of Computer Vision, vol.104, no.2, pp.154–171, 2013.
[30] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," Proc. of European Conference on Computer Vision, vol.8691, pp.346–361, 2014.
[31] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," Proc. of IEEE Computer Vision and Pattern Recognition, pp.3431–3440, 2015.
[32] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," IEEE Trans. Pattern Anal. Mach. Intell., vol.39, no.6, pp.1137–1149, 2017.
[33] A. Vezhnevets and J.M. Buhmann, "Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning," Proc. of IEEE Computer Vision and Pattern Recognition, pp.3249–3256, 2010.
[34] A. Vezhnevets, V. Ferrari, and J.M. Buhmann, "Weakly supervised structured output learning for semantic segmentation," Proc. of IEEE Computer Vision and Pattern Recognition, pp.845–852, 2012.
[35] L. Zhang, Y. Gao, Y. Xia, K. Lu, J. Shen, and R. Ji, "Representative discovery of structure cues for weakly-supervised image segmentation," IEEE Trans. Multimedia, vol.16, no.2, pp.470–479, 2014.
[36] P.O. Pinheiro and R. Collobert, "From image-level to pixel-level labeling with convolutional networks," Proc. of IEEE Computer Vision and Pattern Recognition, pp.1713–1721, 2015.
[37] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," Proc. of International Conference on Learning Representations, 2014.
[38] P. Arbelaez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," Proc. of IEEE Computer Vision and Pattern Recognition, pp.328–335, 2014.
[39] D. Pathak, E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional multi-class multiple instance learning," Proc. of International Conference on Learning Representations, 2015.
[40] D. Pathak, P. Krahenbuhl, and T. Darrell, "Constrained convolutional neural networks for weakly supervised segmentation," Proc. of IEEE International Conference on Computer Vision, pp.1796–1804, 2015.
[41] G. Papandreou, L.-C. Chen, K.P. Murphy, and A.L. Yuille, "Weakly- and semi-supervised learning of a dcnn for semantic image segmentation," Proc. of IEEE International Conference on Computer Vision, pp.1742–1750, 2015.
[42] P. Krahenbuhl and V. Koltun, "Efficient inference in fully connected CRFs with gaussian edge potentials," Advances in Neural Information Processing Systems, 2011.
[43] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and Y.A. L., "Semantic image segmentation with deep convolutional nets and fully connected CRFs," Proc. of International Conference on Learning Representations, 2015.
[44] H. Bilen and A. Vedaldi, "Weakly supervised deep detection networks," Proc. of IEEE Computer Vision and Pattern Recognition, pp.2846–2854, 2016.

[45] A. Diba, V. Sharma, A. Pazandeh, H. Pirsiavash, and L.V. Gool, "Weakly supervised cascaded convolutional networks," Proc. of IEEE Computer Vision and Pattern Recognition, pp.5131–5139, 2017.
[46] D. Kim, D. Cho, and D. Yoo, "Two-phase learning for weakly supervised object localization," Proc. of IEEE International Conference on Computer Vision, pp.3554–3563, 2017.
[47] V. Kantorov, M. Oquab, M. Cho, and I. Laptev, "Contextlocnet: Context-aware deep network models for weakly supervised localization," Proc. of European Conference on Computer Vision, vol.9909, pp.350–365, 2016.
[48] J.T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," Proc. of International Conference on Learning Representations, 2015.
[49] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," Proc. of IEEE Computer Vision and Pattern Recognition, pp.580–587, 2014.
[50] K. Simonyan, A. Vedaldi, and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," Proc. of International Conference on Learning Representations, 2015.
[51] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Is object localization for free? -weakly-supervised learning with convolutional neural networks," Proc. of IEEE Computer Vision and Pattern Recognition, pp.685–694, 2015.

Wataru Shimoda received B.E and M.E degrees from the department of Informatics, the University of Electro-Communications Tokyo, Japan, in 2014 and 2016, respectively. He is now a doctor course student of the University of Electro-Communications and research fellow of Japan Society for the Promotion of Science (JSPS). His research interest includes computer vision and machine learning.

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.