10-2021

A large-scale benchmark for food image segmentation

Xiongwei WU
Singapore Management University, xwwu@smu.edu.sg

Xin FU
Beijing Jiaotong University

Ying LIU
Singapore Management University, yingliu@smu.edu.sg

Ee-peng LIM
Singapore Management University, eplim@smu.edu.sg

Steven C. H. HOI
Singapore Management University, chhoi@smu.edu.sg

See next page for additional authors

A Large-Scale Benchmark for Food Image Segmentation

Xiongwei Wu[1], Xin Fu[2], Ying Liu[1], Ee-Peng Lim[1], Steven C.H. Hoi[1,3], Qianru Sun[1]
[1] Singapore Management University, [2] Beijing Jiaotong University, [3] Salesforce Research Asia
{xwwu,eplim,chhoi,qianrusun}@smu.edu.sg,xinfu@bjtu.edu.cn,rrrainbowly@gmail.com

ABSTRACT

Food image segmentation is a critical and indispensible task for developing health-related applications such as estimating food calories and nutrients. Existing food image segmentation models are underperforming due to two reasons: (1) there is a lack of high quality food image datasets with fine-grained ingredient labels and pixel-wise location masks—the existing datasets either carry coarse ingredient labels or are small in size; and (2) the complex appearance of food makes it difficult to localize and recognize ingredients in food images, e.g., the ingredients may overlap one another in the same image, and the identical ingredient may appear distinctly in different food images.

In this work, we build a new food image dataset FoodSeg103 (and its extension FoodSeg154) containing 9,490 images. We annotate these images with 154 ingredient classes and each image has an average of 6 ingredient labels and pixel-wise masks. In addition, we propose a multi-modality pre-training approach called ReLeM that explicitly equips a segmentation model with rich and semantic food knowledge. In experiments, we use three popular semantic segmentation methods (i.e., Dilated Convolution based [20], Feature Pyramid based [25], and Vision Transformer based [60]) as baselines, and evaluate them as well as ReLeM on our new datasets. We believe that the FoodSeg103 (and its extension FoodSeg154) and the pre-trained models using ReLeM can serve as a benchmark to facilitate future works on fine-grained food image understanding. We make all these datasets and methods public at https://xiongweiwu.github.io/foodseg103.html.

CCS CONCEPTS

• **Computing methodologies** → **Image segmentation**.

KEYWORDS

Datasets, Food Computing, Semantic Segmentation, Deep Learning

ACM Reference Format:
Xiongwei Wu[1], Xin Fu[2], Ying Liu[1], Ee-Peng Lim[1], Steven C.H. Hoi[1,3], Qianru Sun[1]. 2021. A Large-Scale Benchmark for Food Image Segmentation. In *Proceedings of the 29th ACM International Conference on Multimedia (MM '21), October 20–24, 2021, Virtual Event, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3474085.3475201

This work was done during Xin Fu and Ying Liu's internship at Singapore Management University.

MM '21, October 20–24, 2021, Virtual Event, China

ACM ISBN 978-1-4503-8651-7/21/10…$15.00
https://doi.org/10.1145/3474085.3475201

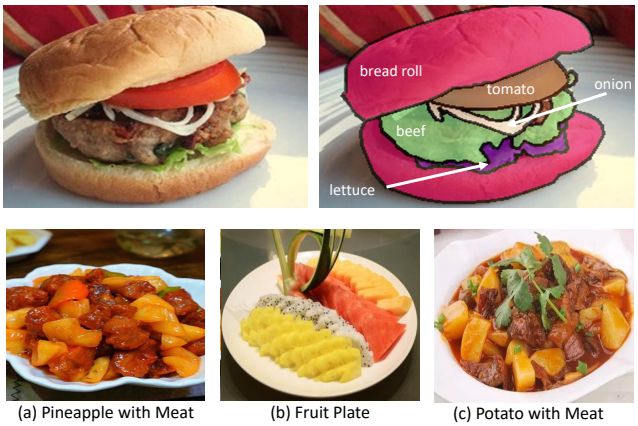


Figure 1: The first row shows a source image and its segmentation masks on our FoodSeg103. The second row shows example images to reveal the difficulties of food image segmentation, e.g., the pineapples in (a) and (b) look different, while the pineapple in (a) and the potato in (c) look quite similar.

1 INTRODUCTION

Food computing has attracted increasing public attention in recent years, as it provides the core technologies for food and health-related research and applications [3, 10, 35, 48]. One of the important goals of food computing is to automatically recognize different types of food and profile their nutrition and calorie values. In computer vision, the related works include dish classification [12, 55, 56], recipe generation [17, 43, 51], and food image retrieval [8, 46].

Most of them focus on representing and analysing the food image as a whole, and do not explicitly localize or classify its individual ingredients—the visible components in the cooked food. We call the former food image classification and the latter food image segmentation. Between the two, food image segmentation is more complex as it aims to recognize each ingredient category as well as its pixel-wise locations in the food image. As shown in Figure 1, given an "hamburger" example image, a good segmentation model needs to recognize and mask out "beef", "tomato", "lettuce", "onion" and "bread roll" ingredients. Food image segmentation is convenient and useful for food quantification, e.g., it can recognize the noodles from different restaurants containing ONE or THREE pieces of marinated pork (differ in calories and fats). Besides, it facilitates a challenging computer vision task.

Compared to semantic segmentation on general object images [5, 20, 25], food image segmentation is more challenging due to the large diversity in food appearances and the often imbalanced distribution of categories of ingredients. First, an ingredient cooked differently can vary a lot visually, e.g., "pineapples" cooked with meat in Figure 1 (a) versus the "pineapples" in a fruit platter in Figure 1

(b). Different ingredients may look very similar, e.g., "pineapples" cooked with meat cannot be easily distinguished from "potatoes" cooked with meat, as shown in Figures 1 (a) and (c) respectively. Second, food datasets usually suffer from imbalanced distribution—both food classes and ingredient classes often exist in long-tailed distributions. This is inevitable due to two reasons: 1) large number of food images are dominated by very few popular food classes while vast majority of food classes are unpopular; and 2) there is a selection bias in the construction of food image collection [49]. We will elaborate the detailed distribution analysis in Section 3.

Existing food image datasets, such as ETH Food101 [2], Recipe1M [45], and Geo-Dish [56], mainly facilitate the research of dish classification or recipe generation. They do not have fine-grained ingredient masks or labels. There are a few public datasets available for food image segmentation [35, 39]. However, their segmentation masks are annotated at dish level only. That is, each mask covers the region of an entire dish instead of that of food ingredients. We elaborate more dataset comparison in Section 3.3.

Dataset contribution: To facilitate fine-grained food image segmentation, we build a large-scale dataset called FoodSeg103, for which we have defined 103 ingredient classes and annotated 7,118 western food images using these labels together with the corresponding segmentation masks. Besides, we annotated an additional set of 2,372 images of Asian food which covers more diverse set of ingredients making these images more challenging than those in the main set (FoodSeg103). For this set, we defined 112 ingredient classes—55% overlap with the ingredient classes of the main set. In total, we annotated 154 classes of ingredients with around 60k masks (in the two datasets). We name the combined dataset as FoodSeg154. During the annotation, we carried out careful data selection, iterative refinement of labels and masks (to be further elaborated in Section 3.2), so as to guarantee high quality labels and masks in the dataset. Our annotation is thus expensive and time-consuming. In experiments, we use FoodSeg103 for in-domain training and testing, and use the additional set in FoodSeg154 for out-domain testing.

Model contribution: The source images of FoodSeg103 are from another existing food dataset Recipe1M [45]—millions of images and cooking recipes, used for recipe generation. Each recipe contains not only "how to cook" but also "what ingredient to use". Auxiliary knowledge is proved useful in data mining [29], and in our work, we leverage these recipe information as auxiliary information to train semantic segmentation models. We call this *multi-modality knowledge transfer* and name our training method ReLeM. Specifically, ReLeM integrates food recipe data, in the format of language embedding, with the visual representation of the food image. In this way, it forces the visual representation of an ingredient appearing in different dishes to have their appearances "connected" in the feature space through a common language embedding (extracted from the ingredient's label and its cooking instructions).

Experiment contribution: We validate our proposed ReLeM model by plugging it into the state-of-the-art semantic segmentation models such as CCNet [20], Sem-FPN [25] and SeTR [60]. In experiments, we compare ReLeM-variants with these baseline models using both convolutional networks and transformer backbones. Our experiments show that ReLeM is generic to be applied into multiple segmentation frameworks, and it helps to achieve significant accuracy improvement when incorporated into the SOTA CNN-based model CCNet. This validates that our knowledge transfer approach works more efficient on stronger models—a characteristic preferred by the multimedia community.

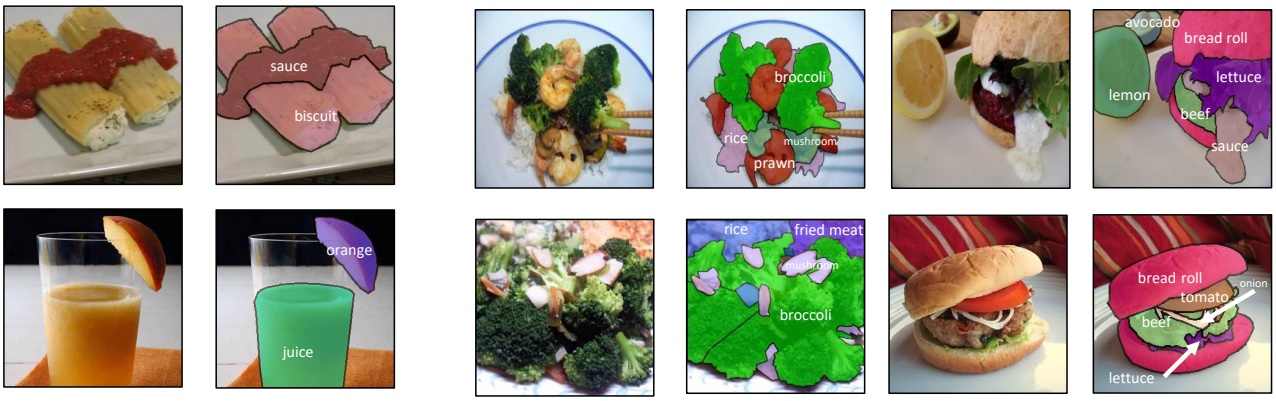
Our contributions are thus three-fold. i) We build a large-scale food image segmentation dataset called FoodSeg103 (and its extension FoodSeg154). It can facilitate a promising and challenging benchmark for the task of semantic segmentation in food images. ii) We propose a knowledge transfer approach ReLeM that utilizes the multi-modality information of recipe datasets. It can be incorporated into different semantic segmentation methods to boost the model performance. iii) We conduct extensive experiments that reveal the challenges of segmenting food on our FoodSeg103 dataset, and validate the efficiency of our ReLeM based on multiple baseline methods.

2 RELATED WORKS

Food Image Datasets. In recent years, the scale of food-related datasets has grown rapidly. For example, Bossard et al [2] built one large-scale food dataset ETH Food101, which contains 101 classes with 1,000 images per class. Matsuda et al. [33] constructed a Japanese food dataset UEC Food100 with 15K images in 100 dish categories. In comparison, ISIA Food500 [38] contains nearly 400k food images in 500 categories, which is the largest food image recognition. In addition, there are also recipe-related datasets. Salvador et al. [45] built the Recipe1M, with nearly 900k images and 1 million recipes, which is widely used in multi-modal learning between images and recipes. Based on Recipe1M, an even larger dataset Recipe1M+ [31] was constructed with more than 13 millions of food images. However, these datasets are mainly built to support food recognition and recipe generation research rather than food image segmentation, so they do not segment food images into multiple masks and labels of ingredient . Food-201 [35] and UEC dataset [14, 39] are the existing datasets for food image segmentation, which contains ~10,000 images with 201/102 categories. Nevertheless, their annotation are limited to dish-wise masks so they cannot be used for ingredient segmentation.

In this paper, we built FoodSeg103 dataset with 7,118 images and more than 40k masks covering 103 food ingredients. In addition, we have collected another image set for Asian food with 2,372 images (for cross-domain evaluation of the models). Combining the main set and the Asian set, we get the FoodSeg154 with nearly 10k images and 60k ingredient masks. To our best knowledge, FoodSeg154 is the first and the largest ingredient-level dataset for fine-grained food image segmentation. Dataset is a key step in developing deep learning based methods. We hope our dataset can inspire more efforts for the task of food image segmentation.

Semantic Segmentation in Images. Deep learning based semantic segmentation is a super hot topic in recent years. Fully convolutional neural network (FCN) [30] is the first semantic segmentation framework based on deep convolutional neural networks. It predicts pixel-wise masks by replacing the fully connected layers with convolution layers and achieves a clear margin of improvement on the model performance. Chen et al. [5] proposed DeepLab which applies dilated convolutional layers in vanilla FCN. The trained model is more effective as the dilation mechanism enlarges the receptive fields while maintaining a high resolution in feature maps. Chen



(a) Easy cases

(b) Hard cases

Figure 2: Foodseg103 examples: source images (left) and annotations (right).

et al [6] proposed the DeepLab v2, which adds an ASPP module to integrate features of different dilation rates. To further include contextual cues, PSPNet [59] proposed a PPM module that aggregates the contextual information using different-size pooling layers. Zhang et al. [57] proposed the layerwise self-distillation method to leverage the different-scale contextual semantics that are represented inside the same network. Wang et al. [53] proposed the non-local networks to encode the relationship between each pair of pixels in the feature map. Zhang et al. [58] proposed an improved version across feature maps. Based on the non-local networks, CCNet [20] adopted a criss-cross attention layer to significantly economize the computation costs of calculating attentions. Most recently, vision transformer (attention-based) [13, 50] was adapted to tackle semantic segmentation problems in [60] recently and achieves state-of-the-art results. In this paper, we conduct extensive experiments on our dataset using three representative semantic segmentation methods: CCNet [20], FPN [25] and SeTR [60]. We also plug the proposed ReLeM into these methods to show its general efficiency.

3 FOOD IMAGE SEGMENTATION DATASET

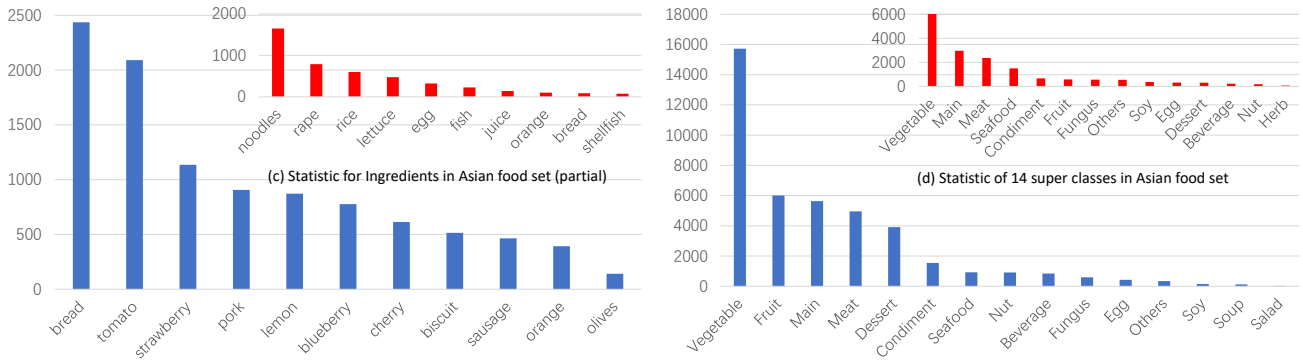
FoodSeg103 is a subset of FoodSeg154, and the latter includes an additional subset of Asian food images and annotations. Some example images and their annotations can be found in Figure 2. In FoodSeg103, we have defined 103 ingredient categories and assigned these category labels as well as the segmentation masks to 7,118 images. The images are from an existing recipe dataset called Recipe1M [45]. For the additional subset in FoodSeg154, we specially collect 2,372 images of Asian food which is of larger diversity than the Western food in FoodSeg103. We use this subset to evaluate the domain adaptation performance of our food image segmentation models. **We release FoodSeg103 to facilitate public research, but currently we cannot make the Asian food set public due to the confidentiality of the images.**

3.1 Collecting Food Images

We use FoodSeg103 as an example to elaborate the dataset construction process. We elaborate the image source, category compilation and image selection as follows. **Source:** We used Recipe1M [31, 45] as our source dataset. This dataset contains 900k images with cooking instructions and ingredient labels, which are used for food image retrieval and recipe generation tasks. **Categories:** First, we counted the frequency of all ingredient categories in Recipe1M. While there are around 1.5k ingredient categories [44], most of them are not easy to be masked out from images. Hence, we kept only the top 124 ingredient categories (with further refinement, this number became 103) and assigned ingredients with the "others" category when they do not fall under the above 124 categories. Finally, we grouped these categories into 14 superclass categories, e.g., "Main" (i.e., main staple) is a superclass category covering more fine-grained categories such as "noodle" and "rice". **Images:** In each fine-grained ingredient category, we sampled Recipe1M images based on the following two criteria: 1) the image should contain at least two ingredients (with the same or different categories) but not more than 16 ingredients; and 2) the ingredients should be visible in the images and easy-to-annotate. Finally, we obtained 7,118 images to annotate masks.

3.2 Annotating Ingredient Labels and Masks

Given the above images, the next step is to annotate segmentation masks, i.e., the polygons covering the pixel-wise locations of different ingredients. This effort includes the mask annotation and mask refinement steps. Each of our images is labeled by one annotator, so there is no inter-annotator agreement. To ensure high quality annotation, a full-time researcher double-checked the labels and corrected errors if any. **Annotation:** We engaged a data annotation company to perform mask annotation, a laborious and painstaking job. For each image, a human annotator first identifies the categories of ingredients in the image, tags each ingredient with the appropriate category label and draws the pixel-wise mask. We asked the annotators to ignore tiny image regions (even if it may contain some ingredients) with area covering less than 5% of the whole image. **Refinement:** After receiving all masks from the annotation company, we further conducted an overall refinement. We followed three refinement criteria: 1) correcting mislabeled data; 2) deleting unpopular category labels that are assigned to less than 5 images, and 3) merging visually similar ingredient categories, such



(a) Statistics of fine-grained ingredient categories (partial)

(b) Statistics of 15 super classes

Figure 3: Category statistics for our FoodSeg103 dataset in (a) and (b), and the Asian food image set (i.e., the additional set in FoodSeg154) in (c) and (d).

as orange and citrus. After refinement, we reduced the initial set of 125 ingredient categories to 103. Figure 5 shows some examples refined by us. The annotation and refinement works took around one year.

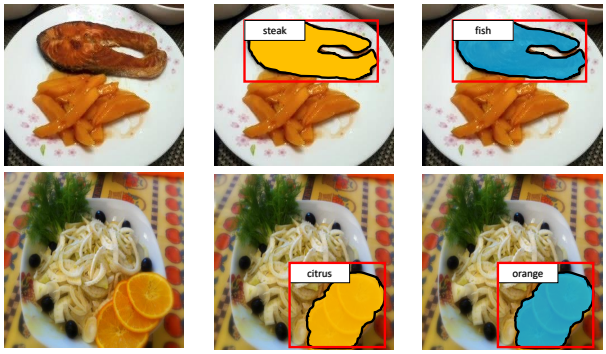
We show some data examples in Figure 2. In Figure 2 (a), we give some easy cases where the boundaries of ingredients are clear and the image compositions are not complex. In Figure 2 (b), we show some difficult cases with overlapped ingredient regions and complex compositions in the images. Figure 3 shows the distributions of fine-grained ingredient categories and superclass categories. Figures 3(a) and 3(c) show partial statistics for small subsets of categories due to page limit. The complete statistics will be published when releasing the dataset.

(a) Source Images (b) Ingredient-level Annotation (c) Dish-level Annotation

Figure 4: Comparison of different annotation styles for masking food images: (a) source images, and (b) ingredient-level annotation (ours), and (c) dish-level annotation [39]. Ingredient-level annotation contains more details.

3.3 Comparing with Food Image Datasets

Food Image Datasets. We summarize the comparison results in Table 1. We only include datasets that are mainly used for food recognition tasks. They contain images and dish-level labels, and therefore



(a) Source Images (b) Before Refinement (c) After Refinement

Figure 5: Examples of dataset refinement. (a) sources images (b) before refinement (wrong or confusing labels exist), and (c) after refinement.

they do not have any ingredient-level annotations. Recipe1M and Recipe1M+ include ingredient labels for each images but not the segmentation masks. Notably, there are some existing datasets for food image segmentation: Food-201 [35], UECFoodPix [14] and UECFoodPixComplete (UECFoodPixComp.) [39]. Below, we compare these datasets with our datasets FoodSeg103 and FoodSeg154 in detail.

Food Image Segmentation Datasets. Food-201 and UEC dataset (UECFoodPix and UECFoodPixComp.) are the public datasets for food image segmentation, with 10k images and 201/102 dish categories. Detailed comparison numbers are given in Table 2. We highlight three advantages of our FoodSeg103 and FoodSeg154: 1) the number of pixel-wise masks of FoodSeg (40k and 60k) is significantly larger than Food-201 and UEC dataset (only 29k and 15k); 2) FoodSeg contains more masks per img (5.9 and 6.3) than Food-201 and UEC dataset (only 2.4 and 1.5). In FoodSeg, only 0.15% of its images have a single mask and nearly 50% images contain more than 3 masks, while in Food-201 and UEC dataset, 57% and 100% images have only a single mask and less than 5% images contain

more than 3 masks; 3) the annotation mask in Food-201 and UEC dataset covers entire dish but not ingredients (dish components), while our FoodSeg154 and FoodSeg103 have ingredient-wise masks, which better capture the characteristic of the food. Illustrative comparisons are given in Figure 4. In Table 2, we present the statistic numbers. FoodSeg103 serves as a more challenging benchmark for semantic segmentation. Moreover, fine-grained ingredient annotations in our datasets are more useful for analyzing food nutrition and estimating calories in health-related applications.

Dataset	Year	Type	#Dish	#Ingr.	Images
PFID [7]	2009	CLS	101	0	4,545
Food50 [21]	2010	CLS	50	0	5,000
Food85 [19]	2010	CLS	85	0	5,500
UEC Food100 [33]	2012	CLS	100	0	14,361
UEC Food256 [23]	2014	CLS	256	0	25,088
Diabetes [1]	2014	CLS	11	0	4,868
ETH Food-101 [2]	2014	CLS	101	0	101,000
UPMC Food-101 [54]	2015	CLS	101	0	90,840
Geo-Dish [56]	2015	CLS	701	0	117,504
UNICT-FD889 [15]	2015	CLS	889	0	3,583
Vireo Food-172 [4]	2016	CLS	172	0	110,241
Food-975 [61]	2016	CLS	975	0	37,785
Food500 [34]	2016	CLS	508	0	148,408
Food11 [47]	2016	CLS	118	0	16,643
Sushi-50 [40]	2019	CLS	50	0	3,963
FoodX-251 [22]	2019	CLS	251	0	158,846
ISIA Food-200 [37]	2019	CLS	200	0	197,323
FoodAI-756 [42]	2019	CLS	756	0	400,000
Recipe1M [45]	2017	Recipe	0	1488	1M
Recipe1M+ [31]	2019	Recipe	0	1488	14M
Food-201 [35]	2015	SEG	201	0	12,093
SUEC Food [16]	2019	SEG	256	0	28,897
UECFoodPix [14]	2019	SEG	102	0	10,000
UECFoodPixComp. [39]	2020	SEG	102	0	10,000
FoodSeg103	2021	SEG	730	103	7,118
FoodSeg154	2021	SEG	730	154	9,490

Table 1: A global view of existing food image datasets. (CLS: no recipe and masks, Recipe: with recipe, SEG: with segmentation masks)

Datasets	# Mask	# Mask Per Img	Image Ratio (%) Over # Mask ≤ 1 Mask	≤ 2 Mask	≤ 3 Mask
Food-201 [35]	29,000	2.4	57.02	82.92	94.93
UECFoodPix [14]	14,011	1.4	100.00	100.00	100.00
UECFoodComp. [39]	16,060	1.6	100.00	100.00	100.00
FoodSeg103	42,097	5.9	0.15	21.90	52.40
FoodSeg154	59,773	6.3	0.15	19.73	47.72

Table 2: Data summary and comparison with existing food image segmentation datasets.

4 FOOD IMAGE SEGMENTATION FRAMEWORK

As shown in Figure 6, our food image segmentation framework contains two modules. One is the *recipe learning module* (ReLeM) to incorporate recipes in the form of language embedding into the visual representation of a food image. We call this approach multi-modality knowledge transfer. In this approach, we explicitly force the visual representations of the same ingredient appearing in different dishes to be "connected" in the feature space through the common language embedding (extracted from the ingredient label and its cooking instructions), so as to handle the high variance of the ingredient appearing in different dishes. The other module of our framework is the *encoder-decoder based image segmentation*. Its encoder is initialized using the one trained by ReLeM, and its decoder is randomly initialized and trained with the segmentation masks. We next introduce the two modules in detail.

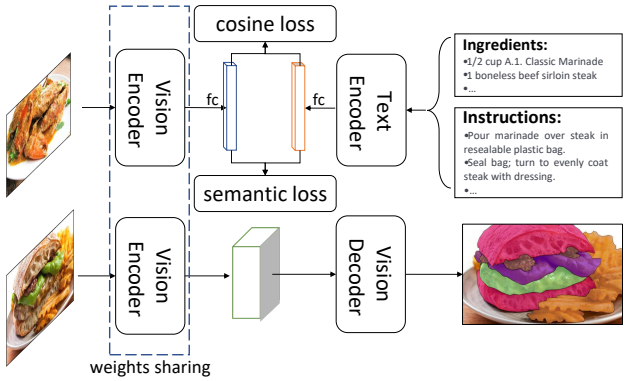


Figure 6: Our food image segmentation framework consists of two modules: Recipe Learning Module (ReLeM) and Image Segmentation Module (Segmenter). For ReLeM, we encode the recipe information into the visual representation of the food image. We deploy the cosine similarity to compute the distance between two distinct-modality models, together with a semantic loss [45]. After training, we use the trained encoder to initialize the encoder of the Segmenter. The decoder of the Segmenter is trained with the segmentation masks from a random initialization.

(a) Source Image (b) Mask Prediction (c) Label Marks

Figure 7: Calculating IoU and Acc, taking the "cake" mask as an example. IoU = $(\frac{TP}{TP+FP+FN})$ and Acc = $(\frac{TP}{TP+FN})$.

Food image segmentation can be viewed as a special type of semantic segmentation [28, 60]. It is more difficult than normal image segmentation due to: 1) the ingredient cooked with different methods can vary a lot by appearances, and 2) ingredient distribution is inevitably long-tailed making the data very sparse for ingredients in the long tail. Given a food image, the Segmenter identifies the ingredient categories and also mask out the corresponding pixels for each category (class). The common metrics for measuring Segmenter's performance include mIoU (mean IoU over each class),

mACC (mean accuracy over all classes) and aAcc (over all pixels), See Figure 7 for more details of IoU and accuracy (Acc) calculation.

4.1 Recipe Learning Module (ReLeM)

Overview. We propose ReLeM to reduce the large intra-variance of ingredients caused by different cooking methods mentioned in the recipes. Specifically, our training method integrates the recipe information into the visual representation of the corresponding image. Assume an ingredient in two different images are cooked in different methods. The visual representations of the ingredients from vision encoder are denoted as v_1 and v_2, where v_1 and v_2 have significant difference in the visual space. ReLeM aims to reduce this difference according to its word embedding of the cooking instructions of the two recipes r_1 and r_2 respectively in the language space.

$$|\phi(v_1|r_1) - \phi(v_2|r_2)| < |\phi(v_1) - \phi(v_2)| \tag{1}$$

where ϕ is the vision decoder in the Segmenter (elaborated in Section 4.2).

Our ReLeM is optimized by using two loss terms: cosine similarity loss between features, and semantic loss (distance) between the text representation t and the visual representation v of the same image:

$$L_{\text{cosine}}((v,t),y) = \begin{cases} 1 - cosine(v,t) & y = 1 \\ max(0, cosine(v,t) - \alpha) & y = -1 \end{cases} \tag{2}$$

$$L_{\text{semantic}}((v,t),u_v,u_t) = \text{CE}(v,u_v) + \text{CE}(t,u_t) \tag{3}$$

where y denotes whether t and v are from the same recipe. u_v and u_t denote the semantic class of u and v respectively, and α is the margin parameter, which is set to 0.1. As Recipe1M does not contain specific semantic labels (i.e., dish names), we define 2,000 semantic labels for it by selecting the most frequent dish names appeared in its recipe titles.

Preprocessing. Each recipe contains ingredients and cooking instructions. Some preprocessing steps are required to encode ingredients and instructions from raw text into the fixed length vectors before they are fed into the text encoder. Specifically, we first extract useful ingredient and instruction texts from the raw recipe data by removing redundant words. For each ingredient, we learn a word2vec [36] representation using a bi-directional LSTM. As the sequence of instructions can be long, it is difficult for LSTM to encode them, due to the gradient vanishing issue. Following a previous work [45], we encode the instructions with a skip-instructions [26] to generate the feature vectors with a fixed length.

Text Encoder. The text encoder is a general module to extract text knowledge from ingredient labels and cooking instructions. We use two types of text encoders: *LSTM-based encoder* and *transformer-based encoder*. For LSTM-based, we use a bi-directional LSTM to encode ingredient features and a LSTM to encode instruction features. For transformer-based model, we use two light-weight transformers, each of which contains 2 transformer layers with 4-head self-attention modules.

Vision Encoder. The vision encoder used in ReLeM aims to extract the visual knowledge from the input image, and the weights will initialize the vision encoder in the segmenter. In this paper, two vision encoders are used: ResNet-50 [18] based on convolutional neural network and ViT-16/B [13] based on vision transformers.

4.2 Image Segmentation Module (Segmenter)

Our framework follows a standard paradigm of semantic segmentation, where the input image is first encoded in a vision encoder, and then goes through a vision decoder to predict masks. Existing segmentation models can be roughly divided into three groups, based on the different designs of encoder and decoder: *Dilation based*, *Feature Pyramid Networks (FPN) based* and *Transformer based*.

Dilation based. Dilation convolution layers aim to enlarge the receptive fields without sacrificing the resolution, as shown in Figure 8 (a). In its decoder, only the last-layer feature maps are used for prediction [6, 20], as shown in Figure 9 (a).

FPN based. FPN integrates feature maps in different layers by the lateral connection. The shallow-layer image representation is enhanced by integrating the feature maps generated in deep layers, as shown in Figure 8 (b). In its decoder, a set of feature pyramids are merged together followed with a mask predictor, as shown in Figure 9 (b).

Transformer based. Transformer is based on attention, which suits semantic segmentation tasks well—-the contextual information is important in segmenting objects. Moreover, the receptive fields can be enlarged via attention mechanism [50, 60]. The transformer based model reshapes the image into a sequence of regions and then encodes them by a sequence of attention modules, as shown in Figure 8 (c). Its decoder predicts segmentation masks on the last-layer feature maps, as shown in Figure 9(c).

In this paper, we conduct experiments using three representative frameworks of these three types, respectively, i.e., CCNet (Dilation) [20], FPN [25] and SeTR (Transformer) [60]. Note that the encoder of Segmenter is pre-trained by our ReLeM. With LSTM and transformer-based text encoding, we arrive at 6 different ReLeM models, i.e., ReLeM-{ CCNet, FPN, SeTR}×({ LSTM, Transformer}). We use the standard pixel-wise cross-entropy loss to optimize segmentation models.

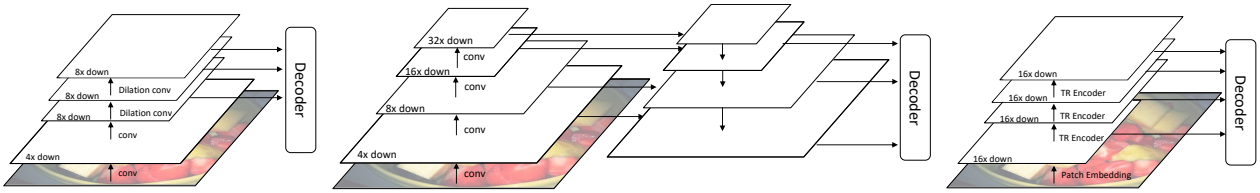
5 EXPERIMENTS

We conduct extensive experiments on our dataset FoodSeg103 and implement our proposed ReLeM by incorporating three baseline methods of semantic segmentation. Below, we first elaborate the experimental settings and the results of an ablation study. Then, we show the performance gaps of the top model in the typical semantic segmentation task and our food image segmentation task. We also evaluate the model adaptability using the Asian food data splits in our FoodSeg154. Lastly, we provide some qualitative results of our best segmentation models.

5.1 Implementation Details

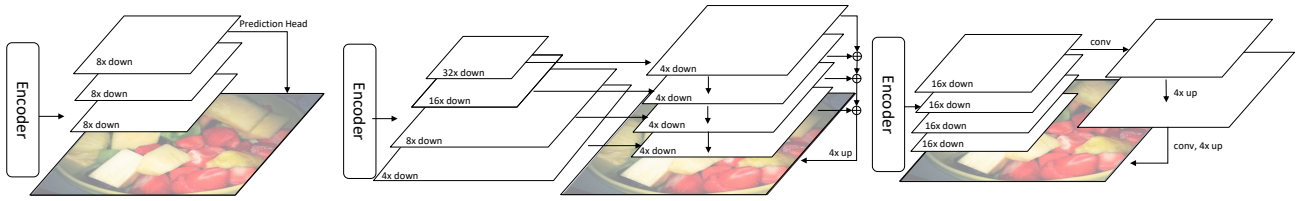
Dataset Settings In our experiments, we use FoodSeg103 for in-domain training and testing, and use the additional Asian food set for out-domain testing. We randomly divide FoodSeg103 dataset into two splits: training set and testing set, according to the 7:3 ratio. Our training set contains 4,983 images with 29,530 ingredient masks, while testing set contains 2,135 images with 12,567 ingredient masks. For ReLeM training, we use the training set of Recipe1M+ to learn the recipe representations (with test images in FoodSeg103 hidden from training).

Segmenter Settings We conduct experiments based on two types of vision encoders: ResNet-50 [18] based on convolutional neural



(a) Dilation Based (b) FPN Based (c) Transformer Based

Figure 8: Different types of encoder for food image segmentation



(a) Dilation Based (b) FPN Based (c) Transformer Based

Figure 9: Different types of decoder for food image segmentation

networks, and ViT-16/B [13] based on vision transformer. ResNet-50 is initialized from the pre-training model on ImageNet-1k [11], which is widely used in multiple vision tasks [6, 27, 41]. ViT-16/B [13] is a transformer-based model, which is initialized from the pre-training model on ImageNet-21k. ViT-16/B contains 12 transformer encoders with 12-head self-attention modules. We use the bilinear interpolation method to reinitialize the pre-trained positional embedding. In this paper, we use three types of segmenters: CCNet [20], FPN [25] and SeTR [60]. CCNet and FPN are based on ResNet-50, while SeTR is based on ViT-16/B. Notably, SeTR extracts feature maps from 12^{th} transformer encoders, followed by two sets of convolution layers for prediction. Other components of the segmentors follow the default settings with random initialization.

ReLeM Settings We use two types of vision encoders in ReLeM: ResNet-50 and ViT-16/B, which follow the same setting as Segmenter. In text preprocessing step, we use the skip-instruction models from the pre-trained weights in [32].

Learning Parameters of Segmenter Each image will be resized into a fixed size of 2049×1024 pixels with a ratio range from 0.5 to 2.0. A 768×768 patch is cropped from the resized images, and random horizontal flipping and color jitter are applied. We trained the models with 80k iterations based on 8 images per batch, and optimized the models by SGD solvers, with a momentum as 0.9 and weight decay as 0.0005. For CCNet and FPN, we set the initial learning rate to 1e-3, while for SeTR we set initial learning rate to 1e-3. According to the general settings [20, 52], the learning rate is decayed by a power of 0.9 according to the polynomial decay schedule. For simplicity, we do not apply hard negative mining during training, and our framework is based on the widely used platform mmsegmentation [9]. All experiments were conducted on 4 Tesla-V100 GPU cards.

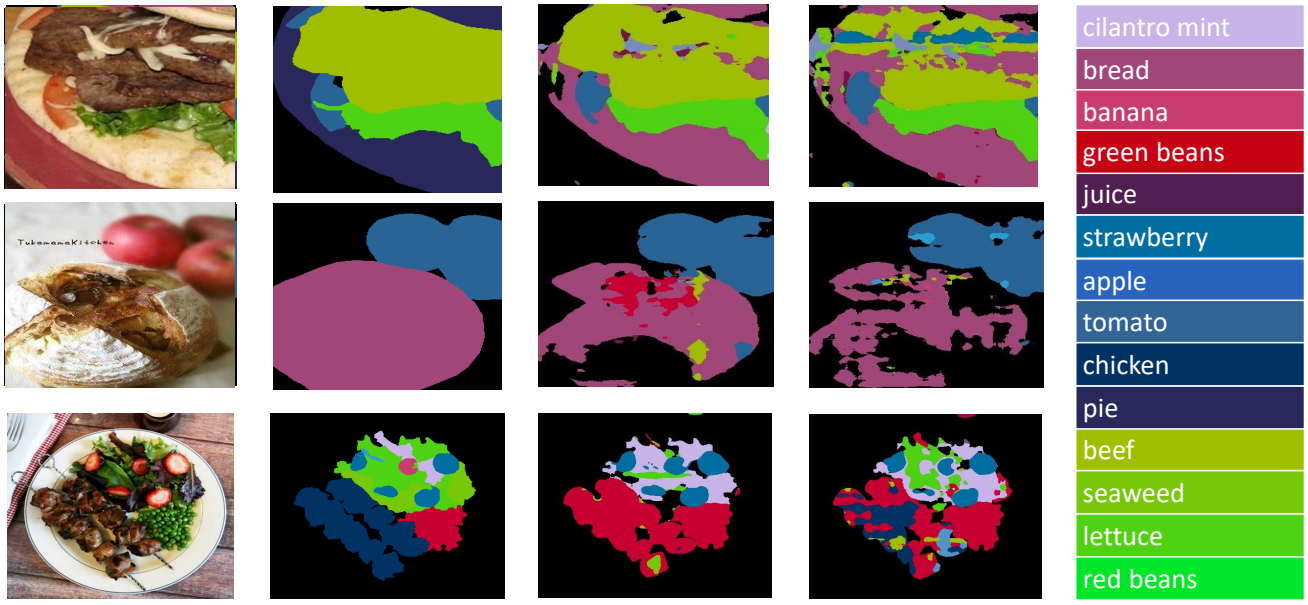
Learning Parameters of ReLeM Each input image are resized into a size of 256×256 pixels and a 224×224 patch is cropped from the resized images as the input of the vision encoder. The model is trained for 720 epochs and each batch contains 160 images. We use Adam solver [24] to optimize the models, with a learning rate of 1e-4, Here we follow a two-stage optimization strategy. We first freeze the weights of the vision encoder and optimize the text encoder. After the text encoder converges, we start to train the vision encoder and freeze the parameters of the text encoder.

5.2 Results and Observations

The experiment results of CCNet, FPN and SeTR on FoodSeg103 are shown in Table 3. The Segmenters of all CCNet, FPN and SeTR achieve significant improvements when incorporating with either LSTM-based or transformer-based ReLeM (1.3%, 1.3% and 2.6% improvement). This confirms that ReLeM is effective in enhancing both convolution based and transformer based semantic segmentation models. Besides, we can see that the performance of using LSTM-based ReLeM is consistently superior than using transformer-based ReLeM across all the model configurations. This is because we used a lightweight Transformer–smaller than LSTM (165M vs 444M).

Methods	mIoU	mAcc	Model Size
CCNet [20] (ResNet-50)	35.5	45.3	381M
ReLeM-CCNet (LSTM)	**36.8**	47.4	381M
ReLeM-CCNet (Transformer)	36.0	46.5	381M
FPN [25] (ResNet-50)	27.8	38.2	218M
ReLeM-FPN (LSTM)	**29.1**	39.8	218M
ReLeM-FPN (Transformer)	28.9	39.7	218M
SeTR [60], (ViT-16/B)	41.3	52.7	723M
ReLeM-SeTR (LSTM)	**43.9**	57.0	723M
ReLeM-SeTR (Transformer)	43.2	55.7	723M

Table 3: Semantic segmentation results of our ReLeM plugged into three baseline methods (on the FoodSeg103 dataset). We implement two variants of ReLeM using LSTM and Transformer, respectively, to encode recipes.



(a) Source Image (b) Ground Truth (c) ReLeM-CCNet (d) CCNet (e) Labels-Masks

Figure 10: Visualization results on FoodSeg103. ReLeM-CCNet can make more accurate predictions.

5.3 Qualitative Examples

In Figure 10, we show some qualitative results of using CCNet and ReLeM-CCNet on the testing set of FoodSed103. The first two rows clearly show that ReLeM-CCNet produces more accurate and detailed predictions than the vanilla CCNet, demonstrating the effectiveness of ReLeM. In the last row, we show a failure case. It is actually a hard example with no clear boundaries among different ingredients.

5.4 Cross-Domain Evaluation

We conduct an out-domain model evaluation using the Asian food data set in FoodSeg154. With the model trained on FoodSeg103, we adapt it to the subset of FoodSeg154, the Asian food data set. Specifically, the Asia food set is evenly divided into the training and testing splits. We fine-tune the trained model on the training set and then run the model on the testing data. In Table 4, we show the performances of three models trained with the following settings: 1) without ReLeM, 2) with ReLeM and 3) with ReLeM and fine-tuned on the training split of the Asian food set. For the first two settings, we only evaluate the 62 classes in Asian food set overlapped with FoodSeg103, and for the last setting, we evaluate 112 classes (all). From the results in Table 4, we observe that using ReLeM consistently outperforms baselines in both cases—with and without model fine-tuning on the training split of Asian food data.

Methods	mIoU	mAcc	aAcc
CCNet	28.6	47.8	78.9
ReLeM-CCNet	29.2	47.5	79.3
CCNet-Finetune	41.3	53.8	87.7
ReLeM-CCNet-Finetune	47.1	59.5	85.5
FPN	21.9	41.7	75.5
ReLeM-FPN	22.9	42.3	77.0
FPN-Finetune	27.1	38.0	82.6
ReLeM-FPN-Finetune	30.8	40.7	78.9

Table 4: Cross-domain adaptation results. We use LSTM based ReLeM.

6 CONCLUSIONS

We construct a large-scale image dataset FoodSeg103 (and its extension FoodSeg154) for food image segmentation research. We use around 10k images and annotate 60k segmentation masks in total, covering highly diverse appearances among 154 ingredients. In addition, we propose a multi-modality based pre-training method ReLeM, and validate its effectiveness by incorporating three baseline semantic segmentation methods and conducting extensive experiments on the FoodSeg103, i.e., using the typical setting, as well as on the FoodSeg154, i.e., using the challenging cross-domain setting.

7 ACKNOWLEDGEMENT

This research is supported by the National Research Foundation, Singapore under its Strategic Capabilities Research Centres Funding Initiative and A*STAR under its AME YIRG grant (Project No. A20E6c0101). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore.

REFERENCES

[1] Marios M Anthimopoulos, Lauro Gianola, Luca Scarnato, Peter Diem, and Stavroula G Mougiakakou. 2014. A food recognition system for diabetic patients

based on an optimized bag-of-features model. *IEEE JBHI* (2014), 1261–1271.
[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101–mining discriminative components with random forests. In *ECCV*. 446–461.
[3] Rebecca G Boswell, Wendy Sun, Shosuke Suzuki, and Hedy Kober. 2018. Training in cognitive strategies reduces eating and improves food choice. *PNAS* (2018), E11238–E11247.
[4] Jingjing Chen and Chong-Wah Ngo. 2016. Deep-based ingredient recognition for cooking recipe retrieval. In *ACMMM*. 32–41.
[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. 2015. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*.
[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. 2017. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI* (2017), 834–848.
[7] Mei Chen, Kapil Dhingra, Wen Wu, Lei Yang, Rahul Sukthankar, and Jie Yang. 2009. PFID: Pittsburgh fast-food image dataset. In *ICIP*. 289–292.
[8] Gianluigi Ciocca, Paolo Napoletano, and Raimondo Schettini. 2017. Learning CNN-based features for retrieval of food images. In *ICIAP*. 426–434.
[9] MMSegmentation Contributors. 2020. MMSegmentation: OpenMMLab Semantic Segmentation Toolbox and Benchmark. https://github.com/open-mmlab/mmsegmentation.
[10] Tilman David and Clark Michael. 2014. Global diets link environmental sustainability and human health. *Nature* (2014), 518–22.
[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*. 248–255.
[12] Lixi Deng, Jingjing Chen, Qianru Sun, Xiangnan He, Sheng Tang, Zhaoyan Ming, Yongdong Zhang, and Tat Seng Chua. 2019. Mixed-dish recognition with contextual relation networks. In *ACMMM*. 112–120.
[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*.
[14] Takumi Ege and Keiji Yanai. 2019. A New Large-scale Food Image Segmentation Dataset and Its Application to Food Calorie Estimation Based on Grains of Rice. In *MADiMa*. 82–87.
[15] Giovanni Maria Farinella, Dario Allegra, and Filippo Stanco. 2014. A benchmark dataset to study the representation of food images. In *ECCV*. 584–599.
[16] Junyi Gao, Weihao Tan, Liantao Ma, Yasha Wang, and Wen Tang. 2019. MUSE-Food: Multi-Sensor-based food volume estimation on smartphones. In *IEEE SmartWorld*. 899–906.
[17] Helena H. Lee, Ke Shu, Palakorn Achananuparp, Philips Kokoh Prasetyo, Yue Liu, Ee-Peng Lim, and Lav R Varshney. 2020. RecipeGPT: Generative pre-training based cooking recipe generation and evaluation system. In *WWW*. 181–184.
[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *CVPR*. 770–778.
[19] Hajime Hoashi, Taichi Joutou, and Keiji Yanai. 2010. Image recognition of 85 food categories by feature fusion. In *ISM*. 296–301.
[20] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. 2019. CCNet: Criss-Cross Attention for Semantic Segmentation. In *ICCV*. 603–612.
[21] Taichi Joutou and Keiji Yanai. 2009. A food image recognition system with multiple kernel learning. In *ICIP*. 285–288.
[22] Parneet Kaur, Karan Sikka, Weijun Wang, Serge J. Belongie, and Ajay Divakaran. 2019. FoodX-251: A Dataset for Fine-grained Food Classification. In *CVPRW*.
[23] Yoshiyuki Kawano and Keiji Yanai. 2014. Automatic expansion of a food image dataset leveraging existing categories with domain adaptation. In *ECCV*. 3–17.
[24] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.
[25] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. 2019. Panoptic feature pyramid networks. In *CVPR*. 6399–6408.
[26] Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-thought vectors. *arXiv preprint arXiv:1506.06726* (2015).
[27] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *NeurIPS*. 1097–1105.
[28] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. 2017. Feature Pyramid Networks for Object Detection. In *CVPR*. 2117–2125.
[29] Zhenguang Liu, Peng Qian, Xiaoyang Wang, Yuan Zhuang, Lin Qiu, and Xun Wang. 2021. Combining Graph Neural Networks with Expert Knowledge for Smart Contract Vulnerability Detection. *TKDE* (2021).
[30] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. In *CVPR*. 3431–3440.
[31] Javier Marin, Aritro Biswas, Ferda Ofli, Nicholas Hynes, Amaia Salvador, Yusuf Aytar, Ingmar Weber, and Antonio Torralba. 2019. Recipe1M+: A Dataset for Learning Cross-Modal Embeddings for Cooking Recipes and Food Images. *TPAMI* (2019), 187–203.
[32] Javier Marín, Aritro Biswas, Ferda Ofli, Nicholas Hynes, Amaia Salvador, Yusuf Aytar, Ingmar Weber, and Antonio Torralba. 2021. Recipe1M+: A Dataset for Learning Cross-Modal Embeddings for Cooking Recipes and Food Images. *TPAMI* (2021), 187–203.
[33] Yuji Matsuda and Keiji Yanai. 2012. Multiple-food recognition considering co-occurrence employing manifold ranking. In *ICPR*. 2017–2020.
[34] Michele Merler, Hui Wu, Rosario Uceda-Sosa, Quoc-Bao Nguyen, and John R. Smith. 2016. Snap, Eat, RepEat: A food recognition engine for dietary logging. In *MADiMa*. 31–40.
[35] Austin Meyers, Nick Johnston, Vivek Rathod, Anoop Korattikara, Alex Gorban, Nathan Silberman, Sergio Guadarrama, George Papandreou, Jonathan Huang, and Kevin P Murphy. 2015. Im2Calories: towards an automated mobile vision food diary. In *ICCV*. 1233–1241.
[36] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).
[37] Weiqing Min, Linhu Liu, Zhengdong Luo, and Shuqiang Jiang. 2019. Ingredient-Guided Cascaded Multi-Attention Network for Food Recognition. In *ACMMM*. 1331–1339.
[38] Weiqing Min, Linhu Liu, Zhiling Wang, Zhengdong Luo, Xiaoming Wei, and Xiaolin Wei. 2020. ISIA Food-500: A Dataset for Large-Scale Food Recognition via Stacked Global-Local Attention Network. In *ACMMM*. 393–401.
[39] Kaimu Okamoto and Keiji Yanai. 2020. UEC-FoodPIX Complete: A Large-scale Food Image Segmentation Dataset. In *ICPRW*. 647–659.
[40] Jianing Qiu, Frank P.-W. Lo, Yingnan Sun, Siyao Wang, and Benny Lo. 2019. Mining Discriminative Food Regions for Accurate Food Recognition. In *BMVC*.
[41] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *NeurIPS*. 91–99.
[42] Doyen Sahoo, Wang Hao, Shu Ke, Xiongwei Wu, Hung Le, Palakorn Achananuparp, Ee-Peng Lim, and Steven C. H. Hoi. 2019. FoodAI: Food Image Recognition via Deep Learning for Smart Food Logging. In *KDD*. 2260–2268.
[43] Amaia Salvador, Michal Drozdzal, Xavier Giro-i Nieto, and Adriana Romero. 2019. Inverse cooking: Recipe generation from food images. In *CVPR*. 10453–10462.
[44] Amaia Salvador, Michal Drozdzal, Xavier Giro-i Nieto, and Adriana Romero. 2019. Inverse Cooking: Recipe Generation From Food Images. In *CVPR*. 10453–10462.
[45] Amaia Salvador, Nicholas Hynes, Yusuf Aytar, Javier Marin, Ferda Ofli, Ingmar Weber, and Antonio Torralba. 2017. Learning cross-modal embeddings for cooking recipes and food images. In *CVPR*. 3020–3028.
[46] Wataru Shimoda and Keiji Yanai. 2017. Learning food image similarity for food image retrieval. In *BigMM*. 165–168.
[47] Ashutosh Singla, Lin Yuan, and Touradj Ebrahimi. 2016. Food/Non-food Image Classification and Food Categorization using Pre-Trained GoogLeNet Model. In *MADiMa*. 3–11.
[48] Quin Thames, Arjun Karpur, Wade Norris, Fangting Xia, Liviu Panait, Tobias Weyand, and Jack Sim. 2021. Nutrition5k: Towards Automatic Nutritional Understanding of Generic Food. In *CVPR*. 8903–8911.
[49] Antonio Torralba and Alexei A Efros. 2011. Unbiased look at dataset bias. In *CVPR*. 1521–1528.
[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *NeurIPS*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.). 5998–6008.
[51] Hao Wang, Guosheng Lin, Steven CH Hoi, and Chunyan Miao. 2020. Structure-Aware Generation Network for Recipe Generation from Images. In *ECCV*. 359–374.
[52] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. 2021. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *ICCV*.
[53] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. 2018. Non-local neural networks. In *CVPR*. 7794–7803.
[54] Xin Wang, Devinder Kumar, Nicolas Thome, Matthieu Cord, and Frederic Precioso. 2015. Recipe recognition with large multimodal food dataset. In *ICME*. 1–6.
[55] Yunan Wang, Jing-jing Chen, Chong-Wah Ngo, Tat-Seng Chua, Wanli Zuo, and Zhaoyan Ming. 2019. Mixed dish recognition through multi-label learning. In *CEAW*. 1–8.
[56] Ruihan Xu, Luis Herranz, Shuqiang Jiang, Shuang Wang, Xinhang Song, and Ramesh Jain. 2015. Geolocalized modeling for dish recognition. *TMM* (2015), 1187–1199.
[57] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xian-Sheng Hua, and Qianru Sun. 2021. Self-Regulation for Semantic Segmentation. In *ICCV*.
[58] Dong Zhang, Hanwang Zhang, Jinhui Tang, Meng Wang, Xian-Sheng Hua, and Qianru Sun. 2020. Feature Pyramid Transformer. In *ECCV*. 323–339.
[59] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. 2017. Pyramid scene parsing network. In *CVPR*. 2881–2890.

[60] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. 2021. Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers. In *CVPR*. 6881–6890.

[61] Feng Zhou and Yuanqing Lin. 2016. Fine-Grained Image Classification by Exploring Bipartite-Graph Labels. In *CVPR*. 1124–1133.