

Singapore Management University

## Institutional Knowledge at Singapore Management University

---

Research Collection School Of Computing and  
Information Systems

School of Computing and Information Systems

---

12-2020

### A study of multi-task and region-wise deep learning for food ingredient recognition

Jingjing CHEN  
*Fudan University*

Bin ZHU  
*City University of Hong Kong*

Chong-wah NGO  
*Singapore Management University, cwngo@smu.edu.sg*

Tat-Seng CHUA  
*National University of Singapore*

Yu-Gang JIANG  
*Fudan University*

Follow this and additional works at: [https://ink.library.smu.edu.sg/sis\\_research](https://ink.library.smu.edu.sg/sis_research)



Part of the [Artificial Intelligence and Robotics Commons](#), and the [Graphics and Human Computer Interfaces Commons](#)

---

#### Citation

CHEN, Jingjing; ZHU, Bin; NGO, Chong-wah; CHUA, Tat-Seng; and JIANG, Yu-Gang. A study of multi-task and region-wise deep learning for food ingredient recognition. (2020). *IEEE Transactions on Image Processing*. 30, 1514-1526.

Available at: [https://ink.library.smu.edu.sg/sis\\_research/6301](https://ink.library.smu.edu.sg/sis_research/6301)

This Journal Article is brought to you for free and open access by the School of Computing and Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Computing and Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email [cherylds@smu.edu.sg](mailto:cherylds@smu.edu.sg).

# A Study of Multi-Task and Region-Wise Deep Learning for Food Ingredient Recognition

Jingjing Chen<sup>1</sup>, Member, IEEE, Bin Zhu, Graduate Student Member, IEEE, Chong-Wah Ngo, Tat-Seng Chua, and Yu-Gang Jiang<sup>2</sup>, Member, IEEE

**Abstract**—Food recognition has captured numerous research attention for its importance for health-related applications. The existing approaches mostly focus on the categorization of food according to dish names, while ignoring the underlying ingredient composition. In reality, two dishes with the same name do not necessarily share the exact list of ingredients. Therefore, the dishes under the same food category are not mandatorily equal in nutrition content. Nevertheless, due to limited datasets available with ingredient labels, the problem of ingredient recognition is often overlooked. Furthermore, as the number of ingredients is expected to be much less than the number of food categories, ingredient recognition is more tractable in the real-world scenario. This paper provides an insightful analysis of three compelling issues in ingredient recognition. These issues involve recognition in either image-level or region level, pooling in either single or multiple image scales, learning in either single or multi-task manner. The analysis is conducted on a large food dataset, Vireo Food-251, contributed by this paper. The dataset is composed of 169,673 images with 251 popular Chinese food and 406 ingredients. The dataset includes adequate challenges in scale and complexity to reveal the limit of the current approaches in ingredient recognition.

**Index Terms**—Food images, Chinese food, ingredient recognition, deep learning.

## I. INTRODUCTION

FOOD log management aims to quantify food consumption and provides services such as advice on weight-loss strategies. The current practice of logging still relies on manual food intake, which is cumbersome. For example, manually inputting the ingredients of a home-cooked dish is required for nutrition estimation. Furthermore, as reported in [1], self-reporting data obtained from unfriendly logging processes often tends to underestimate the actual food intake. With the

Manuscript received January 9, 2020; revised July 30, 2020 and November 1, 2020; accepted December 3, 2020. Date of publication December 23, 2020; date of current version December 31, 2020. This work was supported in part by the Project from the National Science Foundation (NSF) of China under Grant 62072116 and in part by the Research Grants Council of the Hong Kong Special Administrative Region, China under Grant CityU 11203517. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Zhu Li. (Corresponding author: Jingjing Chen.)

Jingjing Chen and Yu-Gang Jiang are with the School of Computer Sciences, Fudan University, Shanghai 200433, China (e-mail: chenjingjing@fudan.edu.cn).

Bin Zhu and Chong-Wah Ngo are with the Department of Computer Science, City University of Hong Kong, Hong Kong.

Tat-Seng Chua is with the School of Computing, National University of Singapore, Singapore 119077.

Digital Object Identifier 10.1109/TIP.2020.3045639



Fig. 1. Variations in visual appearance and composition of ingredients highlight the challenges of food recognition. The first row shows three examples of dishes for the category “scrambled egg & cucumber”, followed by “sour & spicy diced lotus root” and “shredded oyster mushrooms salad” in the second and third rows respectively.

prevalence use of mobile devices, a more convenient way is by taking a picture of a meal for food recognition and logging.

The automatic dietary recognition and assessment have been an active area of research [2]–[7]. These works basically perform dish recognition, and then search for calories and nutrition information of a dish from the food composition table (FCT). For dishes with standardized cooking methods such as fast food, such work-flow is simple and effective. Nevertheless, there remain many categories of dishes without standard cooking methods, food presentation, and composition of ingredients. Figure 1 shows some examples of dishes, where the composition of ingredients within a food category could be diverse. Take the category “shredded oyster mushrooms salad” (last row of Figure 1) for example, there are very few overlaps in ingredients among these dishes except shredded oyster mushrooms. This intuition motivates the studies of ingredient recognition in this paper - a problem deserved more research attention particularly for the large-scale recognition of ingredients from images in the wild.

As observed in Figure 1, the challenges of food recognition come from the large visual variations within the same food category. The variations introduced as a result of different cooking and cutting methods are hard to be tackled by hand-crafted

features such as SIFT [8], HOG [9] and color [10]. Thanks to deep learning [11]–[15], there have been several recent studies [16], [17] that report high accuracy of food recognition of up to 80% on medium scale benchmark datasets, such as Vireo Food-172 [16] and Food-101 [18]. The success of food classification with deep learning techniques has inspired researchers to explore a more challenging problem, i.e., understanding ingredient composition of a dish [16], [19]–[21].

Ingredient recognition is generally a harder problem than food categorization. The size, shape, and color of an ingredient can exhibit large visual differences due to diverse ways of cooking and cutting, in addition to changes in viewpoints and lighting conditions. This paper studies the recognition of ingredients in the domain of Chinese dishes. This domain is particularly challenging because dishes are often composed of a variety of ingredients being fuzzily mixed, rather than separated into different food containers or as non-overlapping food items as frequently seen in Japanese and Western dishes.

This paper describes two methods for ingredient recognition. The methods are not completely new in the literature of food recognition [16], [20]. This paper presents a throughout analysis of both methods, including their strength and limitation in ingredient recognition. The first method is based on multi-task learning that relies on global image features for simultaneous food and ingredient classifications. The motivation is to exploit the mutual relationship between the food category and ingredients for better performance. The key ingredients of a category remain similar despite composing with different auxiliary ingredients. Knowing the food category basically eases the recognition of ingredients. For example, the ingredient “cherry tomatoes” has a higher chance than “pork” to appear in the food “shredded oyster mushroom salad”. Hence, learning ingredients with the food category in mind in principle shall lead to better performance. The second method does not leverage food category information. Ingredient recognition is performed at the image region level. Instead of globally pooling features for recognition, ingredients are first predicted for each local image patch and then pooled across regions as the final recognition result.

This paper also contributes a large dataset, Vireo Food-251, composed of 169,673 images with 251 Chinese food categories and 406 ingredient labels. In terms of the number of food categories, this new dataset is on par with UEC Food-256 [22] and ChineseFoodNet [23] with 208 categories. Note that ingredient labels are not available on both datasets. In the literature, Food-101 [18] also includes ingredient labels. Nevertheless, it is assumed that all dishes under a food category share the same list of ingredients, which makes the dataset inappropriate for ingredient recognition.

We extend the paper by comparing the originally proposed multi-task learning framework in [16] to region-wise ingredient recognition. More in-depth studies, including the issues of image-level versus region-level recognition, single versus multi-scale feature pooling and single versus multi-task learning, are presented with new empirical insights. Furthermore, we extend the original Vireo Food-172 dataset from 172 to 251 food categories and 353 to 406 ingredient labels. The main contributions are the sharing of a large food dataset, and the

comparative studies of various compelling issues in ingredient recognition through the methods of multi-task learning and region-wise recognition. The rest of the paper is organized as follows. Section II reviews related works while Section III introduces the extended dataset. Section IV presents two baselines, i.e., multi-task learning and region-wise multi-label classification, for ingredient recognition. Section V details the performances of two baselines on Vireo Food-251. Finally, Section VI concludes this paper.

## II. RELATED WORK

Food recognition has become a popular research topic in recent years and variants of recognition-centric approaches have been investigated for different food-related applications. These efforts include food quantity estimation based on depth images [3], image segmentation for volume estimation [24], [25], context-based recognition by GPS and restaurant menus [2], [26], [27], taste estimation [28], multi-food recognition [29]–[32], personalized recognition [33], multi-modal fusion [34] and real-time recognition [5]–[7], [35], [36]. This section mainly reviews previous works on food and ingredient recognition.

### A. Food Recognition

The challenge of food recognition comes from visual variations in shape, color and texture layout. These variations are hard to be tackled by hand-crafted features such as SIFT [8], HOG [9] and color [10]. Instead, the features extracted from deep convolutional neural network (DCNN) [11], which is trained on ImageNet [37] and fine-tuned on food images, often exhibit impressive recognition performance [22], [25], [38]–[42]. Combination of multi-modal features sometimes also leads to better recognition performance, as reported in [38], [43]. Recent works mostly focus on researching new architectures for food recognition, such as Wide-slice residual networks [44] and bin-linear CNN models [45]. As reported in [44], the best performances on both UEC Food-100 and Food 101 are achieved by wide-slice residual networks that contain two branches: a residual network branch and a slice branch network with slice convolutional layers. Apart from deep architectures, different learning strategies are also investigated [16], [19], [46]. For example, in [19] and [16], food recognition is formulated as a multi-task learning problem by leveraging ingredient labels or taste labels as supplementary supervised information. By treating ingredients as privilege information, Meng *et al.*, *meng2019learning* propose a cross-modal alignment and transfer network for food recognition. In addition to using static dataset for model training, zero and few shots learning has also started capturing research attention [47], [48].

### B. Ingredient Recognition

Compared to food categories, ingredients exhibit larger visual appearance variations due to different cooking and cutting methods. Labeling of ingredients also poses a higher challenge and only very few datasets [16] are constructed for ingredient recognition. An early work is PFD [49],



Fig. 2. Examples of food categories in VIREO Food-251.

which leverages the result of ingredient recognition for food categorization. In PFD, based upon the appearance of image patches, pixels are softly labeled with ingredient categories. The spatial relationship between pixels is then modeled as a multi-dimensional histogram, characterized by label co-occurrence and their geometric properties such as distance and orientation. With this histogram representation, PFD shows impressive food recognition performance. PFD, nevertheless, is hardly scalable to the number of ingredients. Using only eight categories of ingredients as demonstrated in [49], the histogram already grows up to tens of thousands of dimensions. Other earlier works explore spatial layout [50], feature mining [18] and image segmentation [25] for ingredient or food item recognition. In [50], ingredient regions are detected by shape and texture models, where the shape is based on DPM (deformable part-based model) while the texture is based on STF (semantic texton forest). Similar to PFD [49], the regions are encoded into a histogram modeling spatial relationship between them for food recognition. The spatial relationship is not statistically encoded as in [49], but rather explicit relationships such as “above”, “below”, and “overlapping” are modeled. Such relationships are helpful for recognizing food such as dessert and fast food, but difficult to be generalized such as for Chinese dishes. In [18], an interesting work that mines the composition of ingredients as discriminative patterns is proposed for food classification. A drawback of this approach is the requirement of image segmentation, which is sensitive to parameter settings and can impact recognition performance. As reported in [18], the performance is not better than of DCNN without image segmentation on the Food-101 dataset. Similar to [24], image segmentation is employed in [25], but using a more advanced technique based on conditional random field (CRF) with unary potentials provided by DCNN [51]. The promising performance in segmentation for western food, nevertheless, comes from the price for requiring training labels that need manual segmentation of food items for model learning. For Chinese food, collecting such training labels is extremely difficult, given the fuzzy composition and placement of ingredients as shown in Figure 1.

Ingredient recognition is posed as a multi-label learning problem [16]. More recent works exploit neural networks,

including DCNN and deep Boltzmann machine (DBM), for this problem [16], [52]. To increase the robustness of recognition, multi-task learning, which leverages food category labels as supplementary supervised information, is often employed for simultaneous classification of food and ingredient labels [16], [19]. As the appearance of an ingredient change depending on food preparation, cooking and cutting methods are also explored as supervised information in [20] for ingredient recognition. However, as food preparation is a process, labeling of ingredients with cooking and cutting attributes is complicated and not intuitive. Other supervised information being explored in the literature include restaurant menus using bipartite graph representation [53], cuisine, and course using DBM [52]. Another branch of approaches pose ingredient recognition as cross-modal learning problem [54]–[56]. Specifically, both images and recipes are projected into a joint embedding space for similarity measure. Ingredients are either extracted from the matched recipe of an image [54], [56] or directly predicted from the joint space [57]. However, as the performance is not scalable to large recipe dataset as studied in [58] and cross-modal learning is inherently a “black box” model, the robustness of these approaches is not yet seriously studied.

### III. DATASET

We construct a large food dataset specifically for Chinese dishes, namely Vireo Food-251. Different from other publicly available datasets [18], [29], [59], both food categories and ingredient labels are included. To the best of our knowledge, this is the largest dataset that provides both food categories and ingredient labels.

#### A. Dataset Collection

VIREO Food-251 is extended from the original Vireo Food-172 [16]. With the newly added 79 food categories, the dataset covers the most of popular Chinese dishes. The new dataset is compiled from the top-200 popular food listed on the website “Go Cooking”<sup>1</sup>. The popularity is sorted based on the number of user-uploaded food images. The popular food list is further combined with the original

<sup>1</sup><https://www.xiachufang.com/category/>

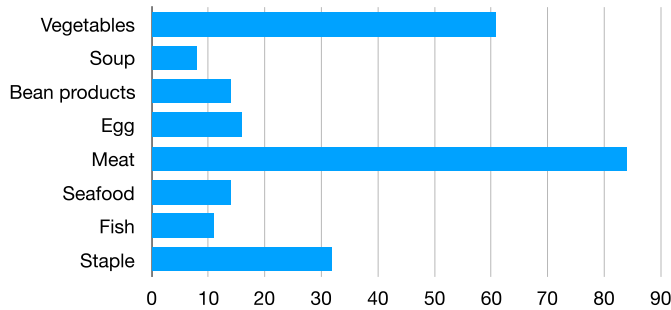


Fig. 3. The distribution of food categories under eight major food groups in Vireo Food-251.

172 categories in the old version, resulting in 251 food categories and 406 ingredient labels. For each newly added food category, a total of 2,000 user-uploaded images were crawled. We manually checked each image by excluding images incorrectly labeled, with multiple dishes, suffered from blurring, or with resolution lower than  $256 \times 256$ .

The 251 categories cover eight major groups of food, as shown in Figure 3. The group *meat* contains the most number of categories, with examples include “braised pork” and “sautéed shredded pork in sweet bean sauce”. On the other hand, there are less than ten categories under the group *soup*, and examples include “lotus root & spare ribs soup” and “crap & tofu soup”. Figure 2 shows some examples of food categories in VIREO Food-251.

### B. Ingredient Labeling

We compiled a list of more than 400 ingredients based on the recipes of 251 food categories. The ingredients range from popular items such as “shredded pork” and “shredded pepper” to rare items such as “codonopsis pilosula” and “radix astragali”. Labeling hundreds of ingredients for over hundred thousands of images could be extremely tedious. First, some ingredients are difficult to be recognized, for example, ingredients under soup or sauce. Second, some ingredients are invisible in flour-made food categories such as dumpling and noodle. Third, certain ingredients such as egg exhibit large visual variations (see Figure 4) due to different ways of cutting and cooking. To address these problems, we label only those ingredients that are visible. In addition, we create additional labels for ingredients with large visual appearance; for example, we have 13 different labels for “egg”, such as “preserved egg slices” and “boiled egg”.

We recruited 10 homemakers who have cooking experience for ingredient labeling. The homemakers were instructed to label only visible and recognizable ingredients. They were also allowed to introduce and annotate new ingredients not in the list, which would be explicitly checked by us. To guarantee the accuracy of labeling, we purposely awarded homemakers with cash bonuses as incentives to provide quality annotation, in addition to the regular payment. For this purpose, we checked a small subset of labels and provided immediate feedback to homemakers such that they were aware of their performance. We spent two months in total to label the whole dataset. By excluding images with no ingredient labels,



Fig. 4. The ingredient “egg” shows large difference in visual appearance across different kinds of dishes.

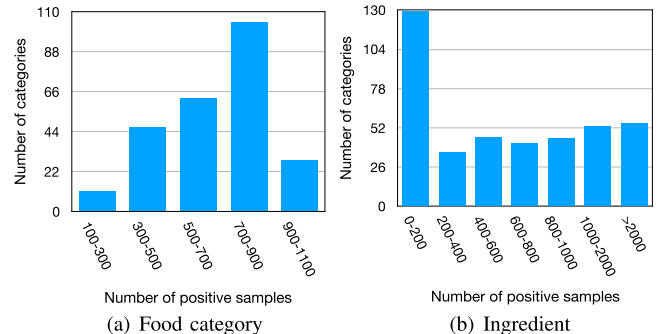


Fig. 5. The distribution of training examples for (a) food categories and (b) ingredient labels.

VIREO Food-251 contains a total of 406 ingredient labels and 169,673 images, with an average of 3 ingredients per image. Figure 5 shows the distribution of positive examples in food and ingredient categories. As observed, the number of training samples is unbalanced. On average, there are 676 positive samples per food category, and 1,196 per ingredient.

## IV. INGREDIENT RECOGNITION

We present two methods for ingredient recognition. The first method is a multi-task model, with two tasks for food and ingredient recognitions [16]. The second model is a single-task model that predicts ingredient labels at local image regions. Both models are based on deep convolutional neural networks (DCNNs).

### A. Multi-Task Learning

The conventional DCNN is an end-to-end system with input as picture and output as the prediction scores of class labels. DCNN models, such as AlexNet [11], VGG [12], and ResNet [13], are trained under the single-label scenario. Specifically, there is an assumption of exactly one label for each input picture. As ingredient recognition is a multi-label problem, i.e., more than one label per image, a different loss function needs to be used for training DCNN. On the other hand, directly revising DCNN with appropriate loss function for ingredient recognition may not yield satisfactory performance, given the varying appearances of an ingredient in different dishes. To this end, we propose to couple food categorization problem, which is a single-label problem, together with ingredient recognition, which is a multi-label problem, for simultaneous learning.

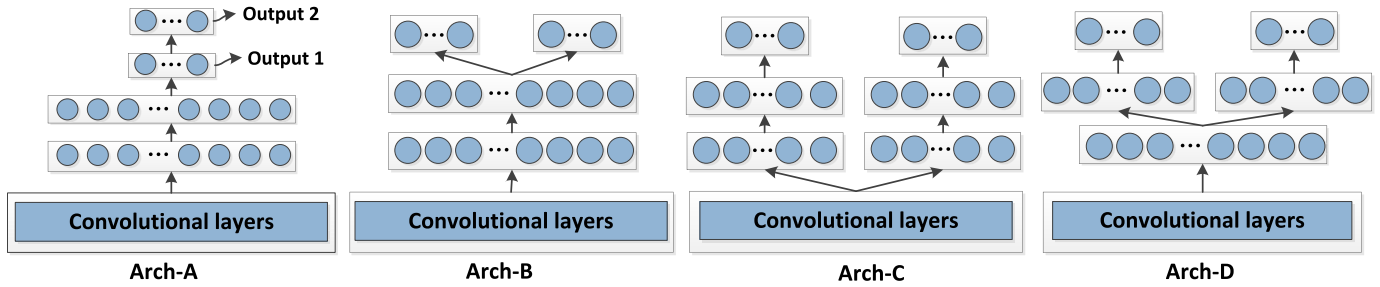


Fig. 6. Four different deep architectures for multi-task learning of food category and ingredient recognition.

We formulate the food categorization and ingredient recognition as a multi-task deep learning problem and modify the architecture of DCNN for our purpose. The modification is not straightforward as it involves two design issues. The first is on whether the prediction scores of both tasks should *directly* or *indirectly* influence each other. Direct influence means that the input of one task is connected as the output of another task. Indirect influence decouples the connection such that each task is on a different path of the network. Both tasks influence each other by updating the shared intermediate layers. The second issue is about the degree in which the intermediate layers should be shared. Ideally, each task should have its own private layer(s) given that the nature of both tasks, single versus multi-labeling, is different. In such a way, the updating of parameters can be done more freely for optimization of individual performance.

Based on the two design issues, we derive four different deep architectures as depicted in Figure 6, respectively name as Arch-A to Arch-D. The first design (Arch-A) considers stacked architecture by placing food categorization on top of ingredient recognition, and vice versa. As the composition of ingredients for different dishes under the same food category can be different, this architecture has the risk that model learning converges slowly as observed in the experiment. The second design (Arch-B) is similar except that indirect influence is adopted and both tasks are at different pathways. Both designs are relatively straightforward to implement by adding additional layers to DCNN. The next two architectures consider the decoupling of some intermediate layers. The third design (Arch-C) allows each task to privately own two intermediate layers on top of the convolutional layers for parameter learning. The last design (Arch-D) is a compromise version between the second and third architectures, by having one shared and one private layer. Arch-D has the peculiarity that the shared layer can correspond to the high or mid-level features common between the two tasks at the early stage of learning, while the private layer preserves the learning of specialized features useful for optimizing the performance of each task.

The architectures are modified from existing deep models, including VGG-16 [12], ResNet-50 [13], ResNet-101 [13], and SENet-154 [60]. In terms of design, the major modification is made on the fully connected layers. As VGG contains two fully connected layers, we modify the fully connected layers to implement all the architectures presented in Figure 6. For the private layers in Arch-D, there are 4,096 neurons for both the

food categorization and ingredient recognition layers. RestNet and SENet, on the other hand, have only one fully connected layer. In this case, only Arch-B can be implemented. Due to the different natures of tasks, we adopt multinomial logistic loss function  $L_1$  for single-label food categorization and cross-entropy as the loss function  $L_2$  for multi-label ingredient recognition. Denote  $N$  as the total number of training images, the overall loss function  $L$  is as following:

$$L = -\frac{1}{N} \sum_{n=1}^N (L_1 + \lambda L_2) \quad (1)$$

where  $\lambda$  is a trade-off parameter. This loss function is also widely used in other works such as [61]. During training, the errors propagated from the two branches are linearly combined and the weights of the convolutional layers shared between the two tasks will be updated accordingly. The updating will subsequently affect the last two layers simultaneously, adjusting the features separately owned by food and ingredient recognition. Let  $\hat{q}_{n,y}$  as the predicted score of an image  $x_n$  for its ground-truth food label  $y$ ,  $L_1$  is defined as follows:

$$L_1 = \log(\hat{q}_{n,y}) \quad (2)$$

where  $\hat{q}_{n,y}$  is obtained from softmax activation function. Furthermore, denote  $p_n \in \{0, 1\}^t$ , represented as a vector in  $t$  dimensions, as the ground-truth ingredients for an image  $x_n$ . Basically  $p_n$  is a binary vector with entries of value 1 or 0 indicating the presence or absence of an ingredient. The loss function  $L_2$  is defined as

$$L_2 = \sum_{c=1}^t p_{n,c} \log(\hat{p}_{n,c}) + (1 - p_{n,c}) \log(1 - \hat{p}_{n,c}) \quad (3)$$

where  $\hat{p}_{n,c}$  denotes the probability of having ingredient category  $c$  for  $x_n$ , obtained through sigmoid activation function.

### B. Region-Wise Ingredient Recognition

The previous section considers the global image feature for multi-label learning, while ignoring regional information. This section introduces region-wise ingredient recognition, as illustrated by the pipeline in Figure 7. Given a food image  $I$ , the feature map (denoted as  $F_I \in \mathbb{R}^{m \times m \times d}$ ), which corresponds to the last convolution layer of DCNN and retains the spatial information of the original image, is extracted from DCNN. The feature map is divided into  $m \times m$  grids, where each grid is represented by a vector of

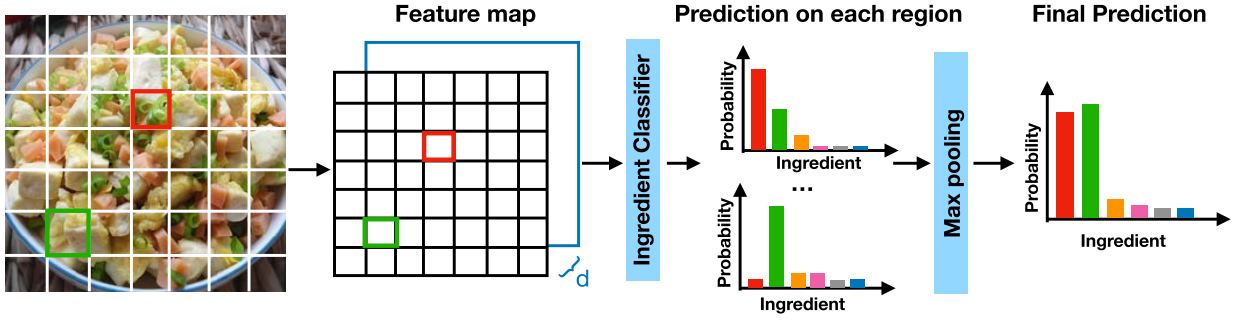


Fig. 7. The pipeline of region-wise ingredient recognition. Given a food image, the feature maps from the last convolutional layer of deep models are extracted for region-wise ingredient classification. Max pooling is performed across different regions to obtain the final predictions.

$d$  dimensions. The value of  $m$  varies depending on the image size. Using VGG as an example,  $m = 14$  if the size of an image is  $448 \times 448$ . In this case, each grid corresponds to a receptive field of  $32 \times 32$  resolution. We denote  $F_i$  as the feature vector for  $i^{\text{th}}$  grid or region, where  $i \in [0, m \times m]$ .

As each grid depicts a small region of the original image, a reasonable assumption is that there is one dominant ingredient per region. Hence, ingredient recognition is performed in a region-wise manner by single-label classification on each grid. The activation function being applied is *softmax* for getting the probability distributions of ingredients, denoted as  $\hat{p}_i \in \mathbb{R}^t$  for  $i^{\text{th}}$  region as follows:

$$\hat{p}_i = \text{softmax}(\mathbf{W}F_i + \mathbf{b}). \quad (4)$$

The learnt transformation matrix is  $\mathbf{W} \in \mathbb{R}^{t \times d}$ ,  $\mathbf{b} \in \mathbb{R}^t$  is the bias terms, and  $t$  is the number of ingredients.

Since each region is associated with the probability distributions of the ingredients, a straightforward way to obtain the image-level labels is by max-pooling over the distributions across regions. Let  $\hat{p}_I$  be the probability distribution of ingredients for image  $I$ . The response of an ingredient indexed by  $j$  element is obtained as follows:

$$\hat{p}_I(j) = \max\{\hat{p}_i(j) |_{i=1}^{m^2}\} \quad (5)$$

where  $m^2$  is the total number of image grids. The loss function is cross-entropy since ingredient recognition is a multi-label classification problem. Denote  $\mathbf{p}_{I_n} \in \{0, 1\}^t$  as the ground-truth ingredients for a food picture  $I_n$ , represented by a binary vector whose elements are either 1 or 0 indicating the presence or absence of a particular ingredient. The loss function  $L$  is defined as

$$L = \frac{1}{N} \sum_{n=1}^N \left( \sum_{j=1}^t \mathbf{p}_{I_n, j} \log(\hat{p}_{I_n, j}) + (1 - \mathbf{p}_{I_n, j}) \log(1 - \hat{p}_{I_n, j}) \right) \quad (6)$$

## V. EXPERIMENTS

The experiments are conducted on the VIREO Food-251 dataset. In each food category, 60% of images are randomly picked for training, while 10% for validation and the remaining 30% for testing. Note that only 385 ingredient categories that have at least 10 training examples are evaluated. As ingredient recognition is a multi-label problem, micro-F1 and macro-F1 that take into account both precision and

TABLE I  
PERFORMANCE COMPARISON AMONG DIFFERENT MULTI-TASK LEARNING ARCHITECTURES FOR INGREDIENT RECOGNITION ON VIREO FOOD 251 DATASET. VGG IS UTILIZED AS THE BACKBONE NETWORK. MICRO-F1 AND MACRO-F1 ARE REPORTED

	Method	Micro-F1 (%)	Macro-F1 (%)
Single-task	VGG	63.99	56.79
Multi-task	Arch-A1	70.81	53.27
	Arch-A2	73.42	58.32
	Arch-B	74.38	59.69
	Arch-C	75.60	61.45
	Arch-D	<b>75.77</b>	<b>61.74</b>

recall of ingredient recognition are employed as evaluation metrics. We split the experiments into two parts to verify the performances of multi-task learning (Section V-A) and region-wise recognition (Section V-B) respectively. The first part aims to evaluate different deep architectures for multi-task learning in comparison to single-task DCNN. The second part aims to demonstrate the merits of region-wise learning for ingredient recognition.

### A. Multi-Task Learning

For multi-task model training, we fix the value of  $\lambda = 0.3$  in Equation 1 for VGG model. When  $\lambda = 0.3$ , the ingredient recognition achieves the best performance on the validation set. As ingredient recognition involves multiple labels, a threshold is required to gate the selection of labels. The threshold is set to be the value of 0.5, following the standard setting when sigmoid is used as the activation function. The learning rate is set to be 0.001 and the batch size to be 64. The learning rate decays when the model reaches a plateau. We first evaluate the performances of different multi-task learning architectures by using VGG as the backbone network. The multi-task learning includes the four deep architectures illustrated in Figure 6. Note that we experiment with two variants of Arch-A, with the layer of food categorization on top of ingredient recognition (Arch-A1) and vice versa (Arch-A2). For comparison, the single task VGG trained with ingredient labels only is utilized as the baseline.

Table I lists the performances of different multi-task architectures for ingredient recognition. Except for Arch-A, all multi-task models exhibit better performance than single-task VGG. As the recognition results for both food and ingredient are imperfect, layer stacking as in Arch-A actually could hurt each other's performance. Specifically, the inaccurate

TABLE II

PERFORMANCE COMPARISON AMONG DIFFERENT MULTI-TASK LEARNING ARCHITECTURES FOR FOOD CATEGORIZATION ON VIREO FOOD 251 DATASET. VGG IS UTILIZED AS THE BACKBONE NETWORK AND AVERAGE TOP-1 AND TOP-5 ACCURACIES ARE REPORTED

	Method	Top-1 (%)	Top-5 (%)
Single-task	VGG	84.46	96.92
Multi-task	Arch-A1	86.41	97.31
	Arch-A2	80.03	89.97
	Arch-B	86.01	97.52
	Arch-C	86.68	<b>97.59</b>
	Arch-D	<b>86.91</b>	97.58

prediction in one task will directly affect the other task. On the other hand, while having separate paths as in Arch-B leads to better performances, the improvement is smaller compared with Arch-C and Arch-D that do not share the same lower layer before the classification layer. Arch-D, which shares one layer while also learning separate layers tailor-made for both tasks, attains the best performance in terms of Micro-F1 and Macro-F1.

Table II lists the performance of food categorization. For multi-task learning, similar trends are observed as ingredient recognition. For top-1 accuracy, the best result is attained by Arch-D while for top-5, the best results are attained by Arch-C. This basically verifies the importance of private layers for both tasks. It is worth to note that, different from ingredient recognition, Arch-A1 performs much better than Arch-A2 for food categorization. The result indicates that recognizing the food category based on the composition of the ingredients is more feasible than inferring ingredients based on the food category. To verify that the improvement is not by chance, we conduct a significance test to compare multi-task (Arch-D) and single-task (VGG) using the source code provided by TRECVID<sup>2</sup>. The test is performed by partial randomization with 100,000 iterations, with the null hypothesis that the improvement is due to chance. At a significance level of 0.05, Arch-D is significantly different from VGG in both food categorization and ingredient recognition by Top-1 accuracy and Macro-F1, respectively. The p-values are close to 0, which rejects the null hypothesis.

To further contrast the performance between single-task and multi-task learning models, Table III lists the ingredients showing large deviations in performances. Basically, for ingredients that are unique for a few food categories, the multi-task learning model performs much better than the single-task learning model. For example, “cordyceps sinensis” only appears in “black chicken soup”, and hence multi-task VGG is able to outperform single-task VGG with a large margin. Another example is “red bean paste” which is unique to the food category “traditional Chinese rice-pudding”, multi-task VGG outperforms single-task VGG by 33.1%. On the contrary, multi-task learning suffers from lower performance when confused by the frequently appearing ingredients. As shown in Table III, ingredients such as “corn block”, “cabbage”, “sliced tomato” are always seen

TABLE III

TEN INGREDIENTS SHOWING LARGE PERFORMANCE DIFFERENCES IN F1 BETWEEN SINGLE-TASK VGG AND MULTI-TASK VGG

Ingredient	Single-task VGG	Multi-task VGG
Cordyceps sinensis	0.000	<b>0.286</b>
Red bean paste	0.000	<b>0.331</b>
Lime leaves	0.076	<b>0.274</b>
Raisin	0.000	<b>0.174</b>
Sliced cherry tomatoes	0.117	<b>0.291</b>
Preserved egg	<b>0.957</b>	0.647
Corn block	<b>0.701</b>	0.612
Cabbage	<b>0.841</b>	0.630
Sliced tomato	<b>0.712</b>	0.507
Loofah section	<b>0.806</b>	0.601

TABLE IV

PERFORMANCES OF INGREDIENT RECOGNITION IN EACH GROUP. THE NUMBER OF INGREDIENT CATEGORIES, AVERAGE AND MEDIAN NUMBERS OF TRAINING IMAGES ARE SHOWN IN THE 2ND, 3RD AND 4TH COLUMNS RESPECTIVELY

Food group	Category #	Average #	Median #	Macro-F1 (%)
Meat	66	1,226	1,226	71.41
Vegetables	146	1,269	689	62.56
Fruits	15	204	40	24.03
Fish	7	1,301	1,008	85.18
Seafood	17	903	596	61.95
Egg	17	1,050	481	67.00
Seasonings	47	2,374	813	46.39
Dry fruits	12	738	413	49.82
Bean products	9	1,375	801	68.71
Others	49	895	455	69.51

in different food categories. Introducing the food category information for multi-task learning will increase the confusion and hence resulted in lower recognition performance. Overall, with multi-task learning, the macro-F1 is boosted from 56.79% to 61.74%, with 285 ingredients showing improvements.

To provide insights on which type of ingredients are difficult to recognize, we divide the ingredients into ten major food groups and report the Macro-F1 of each group in Table IV. As shown, the Macro-F1 for “fish” and “meat” are fairly high due to a sufficient number of training samples. The average numbers of training samples for the ingredient in “meat” and “fish” are 1,226 and 1,301, respectively, which results in high recognition accuracies. On the contrary, the group “fruits” has only 240 training samples on average, which is the fewest among all the groups. The median number is even fewer, which is only 40 as most of the training samples are from the ingredient “pineapple”. As a result, the Macro-F1 of “fruits” is rather low, which is only 24.03%. Despite having 903 training samples on average, the group “seasonings” has the second-lowest Macro-F1, which suggests that recognizing ingredients in the “seasonings” group is relatively challenging compared with the other ingredients.

Figure 8 shows three failure examples of seasoning ingredient recognition. In these examples, seasoning ingredients are “dried chili”, “minced garlic”, “broad bean paste”, “minced ginger” and “minced green onion”. Basically, there are three major reasons for the low performance of seasoning ingredient recognition. First, some seasoning ingredients tend to confuse with each other due to similar appearances. For example, in Figure 9(a), “dried chilli sections” is incorrectly

<sup>2</sup><http://www-nlpir.nist.gov/projects/t01v/trecvid.tools/randomization.testing>



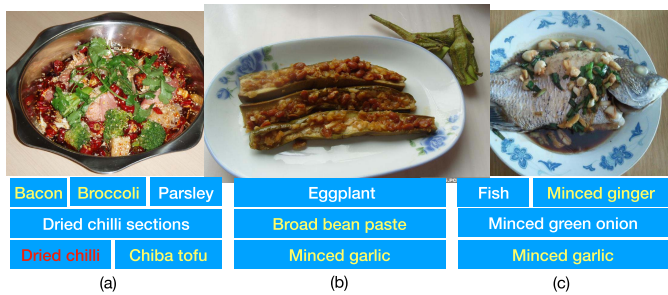


Fig. 8. Failure examples of seasoning ingredients (i.e., “dried chili”, “minced garlic”, “broad bean paste”, “minced ginger”, “minced green onion”) recognition. False positives are marked in red while false negatives are marked in yellow.

predicted as “dried chili”. As “dried chili sections” differs from “dried chili” only in shape, the model tends to confuse them because of occlusions among different ingredients in the dish. Second, seasoning ingredients tend to be small in size and portion, which makes the recognition of “minced garlic” in Figure 9(b)-(c) and “minced ginger” in Figure 9(c) difficult. Such examples are not easy to be recognized even by humans. Third, as the training samples for some seasoning ingredients are not sufficient, the recognition performance is not satisfactory. For example, there are only 18 samples for seasoning ingredient “broad bean pasta”, resulting in incorrect prediction as shown in Figure 9(b).

To validate the the effectiveness of multi-task learning strategy for other backbone networks (e.g., ResNet [13], Inception-V3 [62], SENet-154 [60]), we report the performances of different backbone networks in Table V. As most of the convolutional networks contain only one fully connected layer, hence we implement Arch-B for multi-task learning on different backbone networks. Although Arch-B is not the optimal design for multi-task learning, it still performs better than single-task learning models, which verifies the effectiveness of leverage food category information for ingredient recognition. The general trend is that the deeper a network is, the higher the recognition rate will be. In the case of ingredient recognition, providing food categories as an extra global cue can reduce false positives in ingredient recognition. This advantage does come with the trade-off of introducing more false negatives. As a result, in terms of numerical scores, both single and multi-task learning do not seem to differ too much. We further perform result analysis and notice the following. The false positives introduced by single-task include major ingredients while the false negatives introduced by multi-task are mostly auxiliary ingredients. From the application point of view, auxiliary ingredients have much less impact than major ingredients towards the estimation of nutrition facts. Furthermore, false positives can adversely frustrate user experience in food logging. Hence, multi-task still has its advantage over single-task despite marginal improvement in terms of the numerical score.

### B. Region-Wise Recognition

We then evaluate the performance of region-wise ingredient recognition. Table VI compares the ingredient recognition performance between image-level and region-wise recognition

TABLE V

PERFORMANCE OF MULTI-LABEL INGREDIENT RECOGNITION ON VIREO FOOD-251 DATASET. NOTE THAT ARCH-B IS IMPLEMENTED FOR MULTI-TASK LEARNING

	Method	<i>Micro-F1 (%)</i>	<i>Macro-F1 (%)</i>
Single-task	VGG	63.99	56.79
	ResNet-50	78.59	63.87
	ResNet-101	78.43	65.73
	Inception V3	80.52	66.62
	SENet-154	83.06	72.00
Multi-task	VGG	74.38	59.69
	ResNet-50	78.84	64.11
	ResNet-101	79.02	66.03
	Inception V3	80.71	66.98
	SENet-154	83.27	72.54

TABLE VI

PERFORMANCE OF INGREDIENT RECOGNITION

	Method	<i>Micro-F1 (%)</i>	<i>Macro-F1 (%)</i>
Image-level	VGG	63.99	56.79
	ResNet-50	78.59	63.87
	ResNet-101	78.43	65.73
	Inception V3	80.52	66.62
	SENet-154	<b>83.06</b>	<b>72.00</b>
Region-wise	VGG	66.58	61.53
	ResNet-50	78.77	67.88
	ResNet-101	78.78	68.25
	Inception V3	80.07	67.73
	SENet-154	80.81	68.88

models. Basically, region-wise recognition helps to improve the ingredient recognition performances on all backbone networks except for SENet-154. Since the key idea of SENet is to re-calibrate channel-wise (i.e., region-wise) feature responses by explicitly modeling inter-dependencies between channels (regions), performing region-wise recognition for SENet-154 will harm the dependencies among region features and lead to worse recognition performance. On the contrary, with region-wise recognition, VGG and ResNet-50 further improve macro-F1 by around 5% and 4% respectively. The improvement in terms of micro-F1 is not so obvious, which is around 1%. This is due to the fact that region-wise recognition mostly benefits ingredient categories with a smaller number of training examples. For example, the rare ingredient “cordyceps sinensis” having only 10 training examples improves F1 from 0% to 100%. This is because region-wise recognition, similar to data augmentation, inherently increases the number of training examples. Furthermore, the ingredients that are small in size are less likely to be dominated by other ingredients during feature learning. As a consequence, the contribution of region-wise recognition is more significant for ingredients in small size and with less number of training examples. Figure 9 shows examples to contrast the performance between region-wise and image-level recognitions. Region-wise recognition is robust to size (e.g., “parsley” in Figure 9(a)), cutting method (e.g., “minced ginger” in Figure 9(b)) and training size (e.g., “kiwi fruit” in Figure 9(c)). On the other hand, limited by region size, context information is not fully leveraged. Ingredients such as “crisp fritter” in Figure 9(e) and “yellow peach” are predicted correctly by image-level but not region-wise recognition. Table VII shows the top-10 ingredients that gain the largest improvement due to region-wise recognition.







Input image	Ingredient recognition				Input image	Ingredient recognition			
	Image-level		Region-wise			Image-level		Region-wise	
(a) 	Noodles	Parsley	Noodles	Parsley	(b) 	Tofu blocks	Tofu blocks	Tofu blocks	Tofu blocks
	Beef blocks	Water	Beef blocks	Water		Preserved egg blocks	Preserved egg blocks	Preserved egg blocks	Preserved egg blocks
	Minced green onion		Minced green onion			Minced green onion	Minced green onion	Minced green onion	Minced green onion
	Sliced white radish		Sliced white radish			Minced gingers	Minced gingers	Minced gingers	Minced gingers
(c) 	Shrimp	Flour	Shrimp	Flour	(d) 	Salad dressing	Salad dressing	Salad dressing	Salad dressing
	Green vegetables		Green vegetables			Diced cucumber	Diced carrot	Diced carrot	Diced carrot
	Lemon slices		Lemon slices			Corn	Diced carrot	Corn	Corn
	Kiwi fruit		Kiwi fruit			Green raisins	Green raisins	Green raisins	Green raisins
(e) 	Minced green onion		Minced green onion		(f) 	Yellow peach	Yellow peach	Yellow peach	Yellow peach
	Thin pancake		Thin pancake			White fungus	White fungus	White fungus	White fungus
	Crisp fritter		Crisp fritter			Water	Water	Water	Water
						Wild medlar	Wild medlar	Wild medlar	Wild medlar

Fig. 9. Examples of ingredient recognition. False positives are marked in red while false negatives are marked in yellow. The backbone network is ResNet-50.

TABLE VII

TEN INGREDIENTS SHOWING LARGE PERFORMANCE IMPROVEMENT IN F1 WITH REGION-WISE RECOGNITION MODEL. THE BACKBONE NETWORK IS RESNET-50

Ingredient	Tr. samples #	Image-level	Region-Wise
Cordyceps sinensis	10	0.000	<b>1.000</b>
Milkvetch Root	24	0.000	<b>0.706</b>
Raisin	9	0.000	<b>0.571</b>
Ginseng	12	0.000	<b>0.571</b>
Kiwi fruit	7	0.000	<b>0.444</b>
Vinegar	38	0.222	<b>0.638</b>
Beef ball	13	0.000	<b>0.400</b>
Longan	26	0.000	<b>0.400</b>
Olive pickle	50	0.000	<b>0.400</b>
Sliced radish	7	0.000	<b>0.333</b>

A by-product of the region-wise recognition model is the capability of locating ingredients. We visualize the result in a response map, which is formed by converting the prediction score of an ingredient on an image grid into pixel intensity value. Figure 10 shows the response maps of ingredients. Generally speaking, the better the result of localization is, the higher the prediction accuracy will be.

### C. Discussion

**Why not multi-scale recognition?** As the scales of ingredients change depending on camera-to-dish distance and cutting methods, intuitively region-wise recognition should be benefited from multi-scale processing, as reported in [20]. We input a pyramid of images in multiple resolutions for region-wise recognition. In this way, the receptive field of a grid can spatially extend to a larger scope depending on the resolution of the input image. For example, for an image of size  $448 \times 448$ , each grid in the feature map obtained from VGG corresponds to a receptive field of  $32 \times 32$  image region. By reducing the size of the image to a resolution of  $224 \times 224$ ,

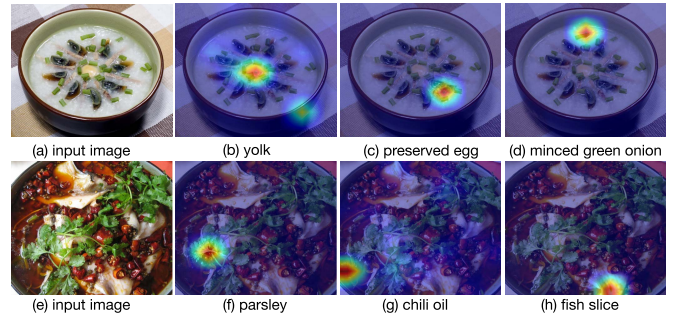


Fig. 10. Ingredient localization: original image (left) and the response maps of three ingredients. The backbone network is ResNet-50.

the receptive field extends to the spatial size of  $64 \times 64$  in the original image before resizing.

The consideration of multi-scale recognition will only introduce minor changes to the original region-wise deep network architecture. Except for region-level pooling that involves ingredient recognition probabilities from multiple scales, the updating of parameters remains the same throughout the learning procedure. Denote  $p_l^i$  as the probability distribution of ingredients at scale  $l$ , max pooling is conducted across different regions and scales as follows:

$$\hat{p}_l(j) = \max\{\max\{\hat{p}_i^l(j)\}_{i=1}^{m^2}\}_{l=1}^L. \quad (7)$$

Basically, the multi-scale design ensures that an ingredient can be adaptively pooled from a region in a particular scale that exhibits the highest possible prediction confidence.

Multi-scale ingredient recognition is performed at two different scales:  $224 \times 224$  and  $448 \times 448$ . Table VIII contrasts the performances between single and multi-scale recognition. Different from the results reported in [20], multi-scale recognition indeed does not show an apparent advantage. In contrast, both micro-F1 and macro-F1 drop for most of the backbones.

TABLE VIII

PERFORMANCE DIFFERENCE BETWEEN SINGLE AND MULTI-SCALE REGION-WISE INGREDIENT RECOGNITION

	Method	<i>Micro-F1 (%)</i>	<i>Macro-F1 (%)</i>
Single-Scale	VGG	66.58	61.53
	ResNet-50	78.77	67.88
	ResNet-101	78.78	68.25
	Inception V3	80.07	67.73
	SENet-154	80.81	68.88
Multi-Scale	VGG	66.53	61.11
	ResNet-50	78.72	68.02
	ResNet-101	78.75	68.31
	Inception V3	78.44	66.43
	SENet-154	76.37	63.21

Our analysis shows that multi-scale recognition boosts the confidence of prediction for both true positive and false negative ingredients. As a consequence, simply using the confidence threshold of 0.5 for selecting ingredients results in slightly lower precision. On the other hand, despite that multi-scale recognition can inherently generate more training samples, its contribution to categories with few training examples is not significant. We argue that a better alternative way is the adaptive fusion of results from multiple scales, rather than simple thresholding for multi labeling, which is beyond the scope of this paper.

**Why not multi-task region-wise recognition?** Region-wise ingredient recognition can be carried out in a multi-task learning fashion together with food recognition. Figure 11 depicts an end-to-end learning architecture, where the two tasks are branched out from the last convolution layer. Intuitively, such architecture might learn to strike a balance between image-level and region-level learning, reaching optimal performance to contextualize ingredient recognition while attending to regional features.

Table IX compares the performance when region-wise recognition is implemented in single and multi-task fashions. As noted, multi-task implementation degrades the recognition rate significantly across different CNN backbones. We attribute the failure to the fact that both tasks indeed perform recognition at different levels of granularities, i.e., image versus region information. Optimizing both tasks based on the architecture in Figure 11 might lead to conflict in learning objectives, resulting in fluctuating performance. For example, the performance of ingredients such as “raisin” and “Ginseng”, which are small in size, decreases when adopting multi-task region-wise recognition. This might be because introducing the image categorization task forces the model to pay more attention to global features which somehow overlooks the small ingredients and harms the regional features optimized for ingredient recognition.

When visualizing the response maps (Figure 12), we also notice that the ingredients cannot be localized as precise as in the single-task model. In some cases, the regions with multiple ingredients are attended, while small-size ingredients are overlooked. Furthermore, the inherent data augmentation in region-wise recognition cannot be effectively leveraged by multi-task learning. As a consequence, the recognition rate for ingredients with a small number of training samples does not improve compared to the single-task model. The result

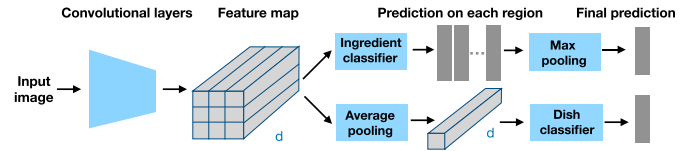


Fig. 11. The pipeline of multi-task region-wise recognition based on ResNet backbone. The model performs region-wise ingredient categorization and image-level food recognition.

TABLE IX

PERFORMANCE DIFFERENCE BETWEEN SINGLE AND MULTI-TASK REGION-WISE INGREDIENT RECOGNITION

	Method	<i>Micro-F1 (%)</i>	<i>Macro-F1 (%)</i>
Single-task region-wise	VGG	66.58	61.53
	ResNet-50	78.77	67.88
	ResNet-101	78.78	68.25
	Inception V3	80.07	67.73
Multi-task region-wise	VGG	63.09	39.76
	ResNet-50	76.61	63.81
	ResNet-101	76.15	61.9
	Inception V3	76.99	60.72

TABLE X

PERFORMANCE OF MULTI-LABEL INGREDIENT RECOGNITION ON UEC FOOD-100

	Method	<i>Micro-F1 (%)</i>	<i>Macro-F1 (%)</i>
Single-task	ResNet-50	61.40	40.19
	ResNet-101	61.75	40.25
	Inception V3	65.34	41.67
	SENet-154	68.95	49.36
Multi-task	ResNet-50	62.88	40.26
	ResNet-101	62.96	40.86
	Inception V3	64.42	42.03
	SENet-154	69.57	49.05

is worse than that of the single-task image-level ingredient recognition, implying that the multi-task model fails in taking advantages of context and region levels information for recognition. More advanced architectures, such as attention branch network that leverages spatial attention [63], are worth further exploration.

**Performance on UEC Food-100 [29] dataset.** We further conduct evaluations on UEC Food-100 dataset. UEC Food-100 is a Japanese food dataset, including 14,361 images from 100 categories of food. [16] labeled this dataset with 190 ingredient classes. By merging the duplicate ingredient labels, we finally obtain 176 ingredients. Basically, similar observations can be found on UEC Food-100 datasets. As shown in Table X, multi-task learning generally improves the performances of ingredient recognition on all backbone models. It is worth noting that due to the lower resolution problem, the ingredient recognition performances on UEC Food-100 are much lower than that on Vireo Food-251.

Table XI further compares the performance between image-level recognition, region-wise recognition and multi-scale region-wise recognition. Similar to the results on Vireo Food-251, region-wise recognition improves the performances in terms of macro-F1. Since region-wise recognition is equivalent to data augmentation, it benefits the recognition of ingredient categories with only a few training samples hence leading to higher macro-F1. However, different to the results on Vireo Food-251, the performances

TABLE XI  
PERFORMANCE OF INGREDIENT RECOGNITION ON UEC FOOD-100

	Method	Micro-F1 (%)	Macro-F1 (%)
Image-level	ResNet-50	61.40	40.19
	ResNet-101	61.75	40.25
	Inception V3	65.34	41.67
	SENet-154	68.95	49.36
Region-wise	ResNet-50	62.37	41.39
	ResNet-101	61.72	41.52
	Inception V3	64.97	42.04
	SENet-154	64.83	43.84
Image-level region-wise	ResNet-50	49.67	28.46
	ResNet-101	53.25	31.11
	Inception V3	54.01	31.61
	SENet-154	64.79	43.44

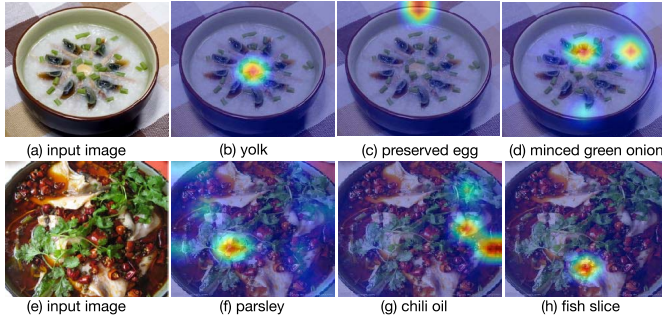


Fig. 12. Ingredient localization with multi-task region-wise recognition model: original image (left) and the response maps of three ingredients. The backbone network is ResNet-50.

of multi-scale recognition are much lower than single-scale recognition. This is due to the reason that the images in UEC Food-100 are in low resolution, resizing the lower resolution images to a higher resolution to perform multi-scale recognition will introduce noise hence leading to worse recognition results.

## VI. CONCLUSION

We have presented a Chinese food dataset, along with two proposed methods for ingredient recognition. The common challenges, regardless of multi-task learning or region-wise recognition, are an unbalanced number of training examples, varying sizes and scales of ingredients under different image capturing conditions. On the other hand, similar to most recognition tasks, the experiments also show a large margin of improvement when deeper networks are employed. Leveraging the food category as a prior, such as in multi-task learning, has advantages for recognizing ingredients that are unique only for a few numbers of food categories. For ingredients frequently appear in different dishes, the performances are either not improved or degraded. Comparing image and region-wise recognitions, the latter improves recognition performance for ingredients in small size and labels with less number of training examples. Region-wise recognition is effective in segregating irrelevant parts of an image from recognition, while augmenting image patches which results in more examples for model training. Nevertheless, as indicated in our result, multi-scale image processing to compensate loss in image context is not helpful for ingredient recognition. Furthermore, image-level food categorization and region-level ingredient recognition, which leverage on different levels of feature

granularities, are conflicting in learning objectives. Optimizing both tasks in a multi-task learning fashion needs more sophisticated network architecture, or otherwise will result in significant performance degradation as shown in our analysis. Future work should pay more attention to adaptive fusion of recognition results from multiple image scales as well as effective leveraging food categorization to contextualize ingredient recognition.

Several research problems can be explored on Vireo Food-251 dataset. First, the dataset is highly unbalanced in the number of training examples for different ingredient labels, ranging from 1 to 32,859 examples. The distribution is long-tail as in real-world scenarios. Solutions such as few-shot learning could be promising for pushing the recognition rate at the tail of the distribution. Second, the co-occurrence probability of ingredients are not random, but follows certain inherent rules in cooking practice. Mining and applying such rules are expected to boost ingredient recognition. Finally, Vireo food-251 can be studied jointly with other datasets for domain adaptation based ingredient recognition. Examples include transferring the model trained by Chinese food to recognize ingredients in Western cuisines with different cooking methods.

## REFERENCES

- [1] A. H. Goris, M. S. Westerterp-Plantenga, and K. R. Westerterp, "Under-eating and underreporting of habitual food intake in obese men: Selective underreporting of fat intake," *Amer. J. Clin. Nutrition*, vol. 71, no. 1, pp. 130–134, Jan. 2000.
- [2] V. Bettadapura, E. Thomaz, A. Parnami, G. D. Abowd, and I. Essa, "Leveraging context to support automated food recognition in restaurants," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Jan. 2015, pp. 580–587.
- [3] M.-Y. Chen *et al.*, "Automatic chinese food identification and quantity estimation," in *Proc. SIGGRAPH Asia Tech. Briefs (SA)*, 2012, p. 29.
- [4] K. Kitamura, T. Yamasaki, and K. Aizawa, "Food log by analyzing food images," in *Proc. 16th ACM Int. Conf. Multimedia*, 2008, pp. 999–1000.
- [5] Z.-Y. Ming, J. Chen, Y. Cao, C. Forde, C.-W. Ngo, and T. S. Chua, "Food photo recognition for dietary tracking: System and experiment," in *Proc. Int. Conf. Multimedia Modeling*. Cham, Switzerland: Springer, 2018, pp. 129–141.
- [6] Y. Kawano and K. Yanai, "FoodCam-256: A large-scale real-time mobile food RecognitionSystem employing high-dimensional features and compression of classifier weights," in *Proc. ACM Int. Conf. Multimedia*, 2014, pp. 761–762.
- [7] K. Aizawa and M. Ogawa, "FoodLog: Multimedia tool for healthcare applications," *IEEE MultimediaMag.*, vol. 22, no. 2, pp. 4–8, Apr. 2015.
- [8] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, Sep. 1999, pp. 1150–1157.
- [9] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 886–893.
- [10] M. A. Stricker and M. Orengo, "Similarity of color images," in *Proc. Storage Retr. Image Video Databases III*, Mar. 1995, pp. 381–392.
- [11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
- [12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: <http://arxiv.org/abs/1409.1556>
- [13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [14] K. Lyu, Y. Li, and Z. Zhang, "Attention-aware multi-task convolutional neural networks," *IEEE Trans. Image Process.*, vol. 29, pp. 1867–1878, 2020.
- [15] Z. Zhang, Y. Xie, W. Zhang, Y. Tang, and Q. Tian, "Tensor multi-task learning for person re-identification," *IEEE Trans. Image Process.*, vol. 29, pp. 2463–2477, 2020.

- [16] J. Chen and C.-W. Ngo, "Deep-based ingredient recognition for cooking recipe retrieval," in *Proc. ACM Multimedia Conf.*, 2016, pp. 32–41.
- [17] N. Martinel, G. Luca Foresti, and C. Micheloni, "Wide-slice residual networks for food recognition," 2016, *arXiv:1612.06543*. [Online]. Available: <http://arxiv.org/abs/1612.06543>
- [18] L. Bossard, M. Guillaumin, and L. Van Gool, "Food-101—mining discriminative components with random forests," in *Proc. Eur. Conf. Comput. Vis. Cham, Switzerland: Springer*, 2014, pp. 446–461.
- [19] X.-J. Zhang, Y.-F. Lu, and S.-H. Zhang, "Multi-task learning for food identification and analysis with deep convolutional neural networks," *J. Comput. Sci. Technol.*, vol. 31, no. 3, pp. 489–500, May 2016.
- [20] J.-J. Chen, C.-W. Ngo, and T.-S. Chua, "Cross-modal recipe retrieval with rich food attributes," in *Proc. 25th ACM Int. Conf. Multimedia*, Oct. 2017, pp. 1771–1779.
- [21] M. Bola nos, A. Ferrà, and P. Radeva, "Food ingredients recognition through multi-label learning," in *Proc. Int. Conf. Image Anal. Process. Cham, Switzerland: Springer*, 2017, pp. 394–402.
- [22] K. Yanai and Y. Kawano, "Food image recognition using deep convolutional network with pre-training and fine-tuning," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jun. 2015, pp. 1–6.
- [23] X. Chen, Y. Zhu, H. Zhou, L. Diao, and D. Wang, "ChineseFoodNet: A large-scale image dataset for chinese food recognition," 2017, *arXiv:1705.02743*. [Online]. Available: <http://arxiv.org/abs/1705.02743>
- [24] M. Puri, Z. Zhu, Q. Yu, A. Divakaran, and H. Sawhney, "Recognition and volume estimation of food intake using a mobile device," in *Proc. Workshop Appl. Comput. Vis. (WACV)*, Dec. 2009, pp. 1–8.
- [25] A. Myers *et al.*, "Im2Calories: Towards an automated mobile vision food diary," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1233–1241.
- [26] R. Xu, L. Herranz, S. Jiang, S. Wang, X. Song, and R. Jain, "Geolocalized modeling for dish recognition," *IEEE Trans. Multimedia*, vol. 17, no. 8, pp. 1187–1199, Aug. 2015.
- [27] Z. Wei, J. Chen, Z. Ming, C.-W. Ngo, T.-S. Chua, and F. Zhou, "DietLens-eout: Large scale restaurant food photo recognition," in *Proc. Int. Conf. Multimedia Retr.*, Jun. 2019, pp. 399–403.
- [28] H. Matsunaga, K. Doman, T. Hirayama, I. Ide, D. Deguchi, and H. Murase, "Tastes and textures estimation of foods based on the analysis of its ingredients list and image," in *Proc. New Trends Image Anal. Process. Workshop. Cham, Switzerland: Springer*, 2015, pp. 326–333.
- [29] Y. Matsuda and K. Yanai, "Multiple-food recognition considering co-occurrence employing manifold ranking," in *Proc. Int. Conf. Pattern Recognit.*, 2012, pp. 2017–2020.
- [30] T. Ege and K. Yanai, "Estimating food calories for multiple-dish food photos," in *Proc. 4th IAPR Asian Conf. Pattern Recognit. (ACPR)*, Nov. 2017, pp. 646–651.
- [31] E. Aguilar, B. Remeseiro, M. Bolanos, and P. Radeva, "Grab, pay, and eat: Semantic food detection for smart restaurants," *IEEE Trans. Multimedia*, vol. 20, no. 12, pp. 3266–3275, Dec. 2018.
- [32] Y. Wang, J.-J. Chen, C.-W. Ngo, T.-S. Chua, W. Zuo, and Z. Ming, "Mixed dish recognition through multi-label learning," in *Proc. 11th Workshop Multimedia Cooking Eating Activities (CEA)*, 2019, pp. 1–8.
- [33] S. Horiguchi, S. Amano, M. Ogawa, and K. Aizawa, "Personalized classifier for food image recognition," *IEEE Trans. Multimedia*, vol. 20, no. 10, pp. 2836–2848, Oct. 2018.
- [34] H. Hoashi, T. Joutou, and K. Yanai, "Image recognition of 85 food categories by feature fusion," in *Proc. IEEE Int. Symp. Multimedia*, Dec. 2010, pp. 296–301.
- [35] Y. Kawano and K. Yanai, "Real-time mobile food recognition system," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2013, pp. 1–7.
- [36] B. V. Resende e Silva, M. G. Rad, J. Cui, M. McCabe, and K. Pan, "A mobile-based diet monitoring system for obesity management," *J. Health Med. Informat.*, vol. 9, no. 2, pp. 1–20, 2018.
- [37] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
- [38] X. Wang, D. Kumar, N. Thome, M. Cord, and F. Precioso, "Recipe recognition with large multimodal food dataset," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jun. 2015, pp. 1–6.
- [39] H. Hassannejad, G. Matriella, P. Ciampolini, I. De Munari, M. Mordonini, and S. Cagnoni, "Food image recognition using very deep convolutional networks," in *Proc. 2nd Int. Workshop Multimedia Assist. Dietary Manage. (MADiMa)*, 2016, pp. 41–49.
- [40] M. Merler, H. Wu, R. Uceda-Sosa, Q.-B. Nguyen, and J. R. Smith, "Snap, eat, Repeat: A food recognition engine for dietary logging," in *Proc. 2nd Int. Workshop Multimedia Assist. Dietary Manage. (MADiMa)*, 2016, pp. 31–40.
- [41] G. Ciocca, P. Napolitano, and R. Schettini, "CNN-based features for retrieval and classification of food images," *Comput. Vis. Image Understand.*, vols. 176–177, pp. 70–77, Nov. 2018.
- [42] S. Jiang, W. Min, L. Liu, and Z. Luo, "Multi-scale multi-view deep feature aggregation for food recognition," *IEEE Trans. Image Process.*, vol. 29, pp. 265–276, Jul. 2020.
- [43] Y. Kawano and K. Yanai, "Food image recognition with deep convolutional features," in *Proc. ACM Int. Joint Conf. Pervas. Ubiquitous Comput. Adjunct Publication UbiComp Adjunct*, 2014, pp. 589–593.
- [44] N. Martinel, G. L. Foresti, and C. Micheloni, "Wide-slice residual networks for food recognition," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 567–576.
- [45] H. Chen, J. Wang, Q. Qi, Y. Li, and H. Sun, "Bilinear CNN models for food recognition," in *Proc. Int. Conf. Digit. Image Comput., Techn. Appl. (DICTA)*, Nov. 2017, pp. 1–6.
- [46] L. Meng *et al.*, "Learning using privileged information for food recognition," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 1–9.
- [47] J. Bootkrajang, J. Chawachat, and E. Trakulsanguan, "Deep-based openset classification technique and its application in novel food categories recognition," in *Proc. Int. Conf. Comput. Recognit. Syst. Cham, Switzerland: Springer*, 2019, pp. 235–245.
- [48] J.-J. Chen, L. Pan, Z. Wei, X. Wang, C.-W. Ngo, and T.-S. Chua, "Zero-shot ingredient recognition by multi-relational graph convolutional network," in *Proc. AAAI*, 2020, pp. 10542–10550.
- [49] S. Yang, M. Chen, D. Pomerleau, and R. Sukthankar, "Food recognition using statistics of pairwise local features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2249–2256.
- [50] H. He, F. Kong, and J. Tan, "DietCam: Multiview food recognition using a multikernel SVM," *IEEE J. Biomed. Health Informat.*, vol. 20, no. 3, pp. 848–855, May 2016.
- [51] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," 2014, *arXiv:1412.7062*. [Online]. Available: <http://arxiv.org/abs/1412.7062>
- [52] W. Min, S. Jiang, J. Sang, H. Wang, X. Liu, and L. Herranz, "Being a supercook: Joint food attributes and multimodal content modeling for recipe retrieval and exploration," *IEEE Trans. Multimedia*, vol. 19, no. 5, pp. 1100–1113, May 2017.
- [53] F. Zhou and Y. Lin, "Fine-grained image classification by exploring bipartite-graph labels," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1124–1133.
- [54] J. Chen, L. Pang, and C.-W. Ngo, "Cross-modal recipe retrieval: How to cook this dish?" in *Proc. Int. Conf. Multimedia Modeling. Cham, Switzerland: Springer*, 2017, pp. 588–600.
- [55] J.-J. Chen, C.-W. Ngo, F.-L. Feng, and T.-S. Chua, "Deep understanding of cooking procedure for cross-modal recipe retrieval," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 1020–1028.
- [56] J.-J. Chen, L. Pang, and C.-W. Ngo, "Cross-modal recipe retrieval with stacked attention model," *Multimedia Tools Appl.*, vol. 77, no. 22, pp. 29457–29473, Nov. 2018.
- [57] A. Salvador *et al.*, "Learning cross-modal embeddings for cooking recipes and food images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3020–3028.
- [58] B. Zhu, C.-W. Ngo, J. Chen, and Y. Hao, "R<sup>2</sup>GAN: Cross-modal recipe retrieval with generative adversarial network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11477–11486.
- [59] M. Chen, K. Dhingra, W. Wu, L. Yang, R. Sukthankar, and J. Yang, "PFID: Pittsburgh fast-food image dataset," in *Proc. 16th IEEE Int. Conf. Image Process. (ICIP)*, Nov. 2009, pp. 289–292.
- [60] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
- [61] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1988–1996.
- [62] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.
- [63] H. Fukui, T. Hirakawa, T. Yamashita, and H. Fujiyoshi, "Attention branch network: Learning of attention mechanism for visual explanation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10705–10714.



**Jingjing Chen** (Member, IEEE) received the Ph.D. degree in computer science from the City University of Hong Kong in 2018. She is currently a pre-tenured Associate Professor with the School of Computer Science, Fudan University. Before joining Fudan University, she was a Postdoctoral Research Fellow with the School of Computing, National University of Singapore. Her research interests include diet tracking and nutrition estimation based on multi-modal processing of food images, including food recognition and cross-modal recipe retrieval.



**Bin Zhu** (Graduate Student Member, IEEE) received the B.Sc. degree from Southeast University, Nanjing, China, in 2015, and the M.Sc. degree from Zhejiang University, Hangzhou, China, in 2018. He is currently pursuing the Ph.D. degree with the VIREO Group, Department of Computer Science, City University of Hong Kong. His research interests include diet tracking, generative model and multimedia analysis, including food recognition, cross-modal recipe retrieval, nutrition estimation, and image generation.



**Chong-Wah Ngo** received the B.Sc. and M.Sc. degrees in computer engineering from Nanyang Technological University, Singapore, and the Ph.D. degree in computer science from The Hong Kong University of Science and Technology (HKUST), Hong Kong. He is currently a Professor with the Department of Computer Science, City University of Hong Kong, Hong Kong. Before joining the City University of Hong Kong, he was a Postdoctoral Scholar with the Beckman Institute, University of Illinois at Urbana-Champaign (UIUC), Urbana, IL,

USA. He was also a Visiting Researcher with Microsoft Research Asia, Beijing, China. His research interests include large-scale multimedia information retrieval, video computing, multimedia mining, and visualization. He was the Conference Co-Chair of the ACM International Conference on Multimedia Retrieval 2015 and the Pacific Rim Conference on Multimedia 2014. He also served as the Program Co-Chair for ACM Multimedia Modeling 2012 and ICMR 2012. He was the Chairman of ACM (Hong Kong Chapter) from 2008 to 2009. He was an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA (2011–2014).



**Tat-Seng Chua** received the Ph.D. degree from the University of Leeds, U.K. He is the KITHCT Chair Professor with the School of Computing, National University of Singapore, where he was the Acting and Founding Dean of the School from 1998 to 2000. His main research interests include multimedia information retrieval and social media analytics. In particular, his research focuses on the extraction, retrieval, and question-answering (QA) of text and rich media arising from the Web and multiple social networks. He is the Co-Director of NEXt, a joint center between NUS and Tsinghua University, to develop technologies for live social media search. He is the 2015 winner of the prestigious ACM SIGMM Award for Outstanding Technical Contributions to Multimedia Computing, Communications, and Applications. He is the Chair of Steering Committee of the ACM International Conference on Multimedia Retrieval (ICMR) and Multimedia Modeling (MMM) conference series. He is also the General Co-Chair of ACM Multimedia 2005, ACM CIVR (now ACM ICMR) 2005, ACM SIGIR 2008, and ACM Web Science 2015. He serves on the editorial boards of four international journals. He is the Co-Founder of two technology startup companies in Singapore.



**Yu-Gang Jiang** (Member, IEEE) received the Ph.D. degree in computer science from the City University of Hong Kong. He is currently a Professor with the School of Computer Science and the Vice Director of the Shanghai Engineering Research Center for Video Technology and System, Fudan University, China. His Laboratory for Big Video Data Analytics conducts research on all aspects of extracting high-level information from big video data, such as video event recognition, object/scene recognition, and large-scale visual search. Before joining Fudan University in 2011, he spent three years at Columbia University. His work has led to many awards, including “Emerging Leader in Multimedia” Award from IBM T. J. Watson Research in 2009, the Early-Career Faculty Award from Intel and China Computer Federation, the 2014 ACM China Rising Star Award, the 2015 ACM SIGMM Rising Star Award, and the Research Award for outstanding young researchers from NSF China.