Few-shot Food Recognition via Multi-view Representation Learning

SHUQIANG JIANG and WEIQING MIN, Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences
YONGQIANG LYU, Qingdao KingAgroot Precision Agriculture Technology Co., Ltd
LINHU LIU, Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences

This article considers the problem of few-shot learning for food recognition. Automatic food recognition can support various applications, e.g., dietary assessment and food journaling. Most existing works focus on food recognition with large numbers of labelled samples, and fail to recognize food categories with few samples. To address this problem, we propose a Multi-View Few-Shot Learning (MVFSL) framework to explore additional ingredient information for few-shot food recognition. Besides category-oriented deep visual features, we introduce ingredient-supervised deep network to extract ingredient-oriented features. As general and intermediate attributes of food, ingredient-oriented features are informative and complementary to category-oriented features, and thus they play an important role in improving food recognition. Particularly in few-shot food recognition, ingredient information can bridge the gap between disjoint training categories and test categories. To take advantage of ingredient information, we fuse these two kinds of features by first combining their feature maps from their respective deep networks and then convolving combined feature maps. Such convolution is further incorporated into a multi-view relation network, which is capable of comparing pairwise images to enable fine-grained feature learning. MVFSL is trained in an end-to-end fashion for joint optimization on two types of feature learning subnetworks and relation subnetworks. Extensive experiments on different food datasets have consistently demonstrated the advantage of MVFSL in multi-view feature fusion. Furthermore, we extend another two types of networks, namely, Siamese Network and Matching Network, by introducing ingredient information for few-shot food recognition. Experimental results have also shown that introducing ingredient information into these two networks can improve the performance of few-shot food recognition.

CCS Concepts: • **Computing methodologies → Image representations**; **Object recognition**;

Additional Key Words and Phrases: Food recognition, few-shot learning, visual recognition, deep learning

This work was supported in part by the National Natural Science Foundation of China under Grants No. 61532018, No. 61972378, and No. U19B2040, in part by Beijing Natural Science Foundation under Grant No. L182054, in part by National Program for Special Support of Eminent Professionals and National Program for Support of Top-notch Young Professionals

Authors' addresses: S. Jiang, Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, No.6 Kexueyuan South Road, Beijing, Beijing, 100190, China; email: sqjiang@ict.ac.cn; W. Min, Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, No.6 Kexueyuan South Road, Beijing, China; email: weiqingmin@ict.ac.cn; Y. Lyu, Shandong Reebow Automation Equipment Co.LTD. Qingdao Branch, Room 1901, building 5, Tianan Digital City, Chengyang District, Qingdao, Shandong, China; email: lvyq@reebow.cn; L. Liu, Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, No.6 Kexueyuan South Road, Beijing, China; email: linhu.liu@vipl.ict.ac.cn.

1551-6857/2020/07-ART87 $15.00
https://doi.org/10.1145/3391624

ACM Reference format:
Shuqiang Jiang, Weiqing Min, Yongqiang Lyu, and Linhu Liu. 2020. Few-shot Food Recognition via Multi-view Representation Learning. *ACM Trans. Multimedia Comput. Commun. Appl.* 16, 3, Article 87 (July 2020), 20 pages.
https://doi.org/10.1145/3391624

1 INTRODUCTION

Food recognition has received a significant amount of attention in various fields, such as computer vision [9], data mining [4], and multimedia communities [26, 58], motivated by many applications in automated food monitoring and dietary management [1, 13], food trend and popularity analysis [2], smart home [66], and food safety [38]. For example, people's diet and eating behavior have been shown to affect their health issues [42]. This fact has fostered the emergence of many approaches to monitor diet. With the fast development of mobile devices, more and more dietary management systems resort to vision-based methods [5, 43]. One necessary and important step is to automatically recognize the type of food displayed in the image. Another example is that food image recognition is a key enabler for many smart home applications such as smart kitchen and smart personal nutrition log [66].

There are more than 8,000 food categories according to Wikipedia [8]. Like other object categories, there is a long-tailed distribution for real-world food data, where many categories have few food samples. For example, when you search some food names, such as "Wagafi bread" and "Babute" using existing popular search engines, such as Google and Bing, very few relevant images are returned. In other words, we can only obtain few examples for these food categories. A robust food recognition system not only recognizes usual food images but also unusual ones. However, existing methods for food recognition need large-scale labeled samples for effective model training [9, 40]. As a result, they cannot handle food categories with few samples. To solve this problem, in this article, we focus on few-shot learning for food recognition, which aims to recognize novel visual food categories from few examples.

There has been a recent resurgence of interest in one/few-shot learning [10, 19, 34, 51, 56, 57, 60]. Various methods, such as Matching Network [60], Prototypical Network [56], and Relation Network [57] have been proposed for few-shot learning. These methods are applied to different domains, such as alphabet recognition and general object recognition. However, besides category information, most existing few-shot learning methods do not explore other types of context information, such as rich attributes and other side information, to enhance the performance of few-shot learning. In addition, as far as we know, there is no work on few-shot learning for food recognition.

Few-shot learning for food recognition is not trivial. The challenges derive from three-fold: First, food image recognition belongs to fine-grained classification [18, 65]. Similarly, food image recognition encounters the same problem as fine-grained classification, such as subtle differences among different food categories. In addition, we cannot directly use existing fine-grained classification methods for food recognition. Existing fine-grained classification methods generally first discover common semantic parts (such as head and breast in the bird dataset), and then fuse features from both global object and semantic parts as final representation. However, the concept of common semantic parts for fine-grained classification methods does not exist in food images. Second, food images do not have any distinctive spatial layout in many cases. Although some food categories such as fruits and hamburgers have regular shapes, many food dishes are lack of rigid structures. Third, few-shot learning for food recognition brings new challenges, such as how to utilize limited samples from some categories to train a robust food recognition model.

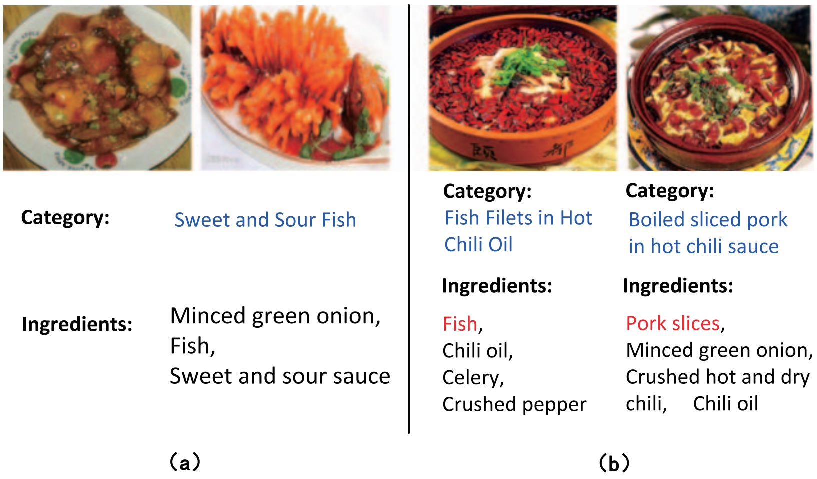


Fig. 1. Examples of food images and associated ingredients from VIREO Food-172.

In many recipe-sharing websites, there are also associated ingredients available. Ingredients can be one constituent part of food as intermediate attributes. It plays an important role in food recognition. For example, Figure 1 shows two groups of food images. In Figure 1(a), these two images from the same category have larger difference in the visual appearance. However, if we consider their ingredient attributes "Minced green onion," "Fish," and "Sweet and sour sauce," and combined the visual representation supervised by ingredient attributes with ones supervised by the category, then the probability of these two images belonging to the same category increases. In Figure 1(b), it is difficult to distinguish between two categories from these two images. However, it is easier to distinguish between them from different ingredient attributes "Fish" and "Pork slices," which are their main characteristics of food categories. If we can use these ingredients to learn to localize relevant image regions, then the probability of these two images belonging to the same category probably decreases. Therefore, ingredient information provides discriminative information for food recognition. Particularly for few-shot food recognition, there are many categories with few samples. As general food attributes, ingredients can serve as important complementary information to improve the performance of few-shot food recognition, and also build the connection between disjoint food categories.

Taking these factors into consideration, we propose a novel Multi-view Few-shot Learning (MVFSL) framework to exploit rich food ingredients for few-shot learning in the food domain. As shown in Figure 2, MVFSL mainly consists of three components: (a) Category and ingredient oriented feature learning; (b) Multi-view feature map fusion; and (c) Multi-view relation learning. Particularly, MVFSL first extracts feature maps from both category-supervised deep network and ingredient-supervised one, respectively. As mid-level attributes, ingredient-oriented features are capable of capturing other details of food, which are complementary to category-oriented features. More specifically, the feature maps of a deep convolutional layer tend to be selective of visual concepts [63]. Therefore, we deem feature maps from ingredient-supervised deep networks focus on salient image regions, which are different from category-oriented feature maps. Then, multi-view feature map fusion is conducted to fuse these two types of feature maps. It first combines extracted feature maps from their respective deep networks, and then convolves combined feature maps. Such convolution is finally involved in multi-view relation learning, which is used to compare pairwise images for metric learning. In multi-view relation learning, a

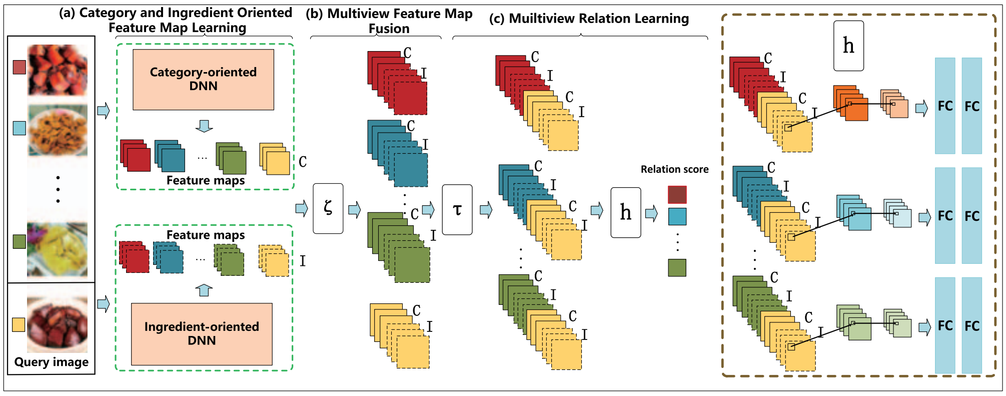

(a) Category and Ingredient Oriented Feature Map Learning
(b) Multiview Feature Map Fusion
(c) Muiltiview Relation Learning
Category-oriented DNN
Feature maps
C
Feature maps
I
Ingredient-oriented DNN
Query image
ζ
τ
h
Relation score
C
I
h
FC
FC
FC
FC
FC
FC

Fig. 2. The proposed architecture for Multi-View Few-Shot Learning (MVFSL), which mainly consists of three components: (a) Category and ingredient oriented feature learning, which is mainly to extract category oriented feature maps C and ingredient oriented feature maps I from Deep Neural Networks (DNN); (b) Multi-view feature map fusion, which can combine different types of feature maps via ζ and then conduct the convolution on combined feature maps, which is tightly associated with (c) Multi-view relation learning, which first combines image-pair features via τ, and then obtains the image-level relation score via h for prediction. In our work, τ is used to concatenate two feature maps according to certain dimension. h generally consists of some convolutional blocks and full-connected layers, and thus can be used for both combined feature maps within each image for multi-view feature fusion and feature maps between two images for image-level relation-score calculation.

multi-view relation network with convolutions and full-connected layers is utilized to apply the convolution to both combined feature maps within each image for multi-view feature fusion and feature maps between two images for image-level fine-grained feature learning. Furthermore, MVFSL can be trained end-to-end to enable joint optimization on different subnetworks.

We conduct comprehensive experimental evaluation on various food benchmarks including western food datasets, such as Food-101 [9] and eastern food datasets, such as VIREO Food-172 [11] and ChineseFoodNet [12]. The experimental results demonstrate the effectiveness of MVFSL in multi-view feature fusion. Furthermore, we extend another two few-shot learning networks including Siamese Network and Matching Network by introducing ingredient information for few-shot food recognition. The experimental results also demonstrate the advantage of these two few-shot learning methods using ingredient information.

The contributions of our article can be summarized as follows:

- To the best of our knowledge, this is the first attempt to apply few-shot learning for food recognition, where rich ingredient information is utilized to improve the performance of few-shot food recognition.
- We propose a Multi-View Few-Shot Learning (MVFSL) framework to exploit rich food ingredients for few-shot food recognition. In MVFSL, multi-view feature map fusion is developed to effectively fuse both ingredient-oriented features and category-oriented ones via the convolution on fused feature maps from multi-view relation subnetwork learning. Furthermore, MVFSL can be trained in an end-to-end manner to enable joint optimization on different subnetworks.
- We conduct comprehensive experimental evaluation on various food benchmarks and experimental results verify the effectiveness of MVFSL. Furthermore, we extend another two few-shot learning methods by introducing ingredient information for few-shot food

recognition. The experimental results again demonstrate the advantage in exploiting ingredient information. The source code can be available[1].

The rest of this article is organized as follows. Section 2 elaborates the proposed Multi-View Few-Shot Learning (MVFSL) framework, where three components are introduced in details, respectively. Section 3 introduced another two types of extended few-shot learning methods using ingredient information, namely, Siamese Network and Matching Network, respectively. Experimental results and analysis are reported in Section 4. Section 5 reviews related work. Finally, we conclude the article and give future work in Section 6.

2 MULTI-VIEW FEW-SHOT LEARNING (MVFSL)

As shown in Figure 2, MVFSL mainly consists of three parts: (a) Category and ingredient oriented feature learning; (b) Multi-view feature map fusion; and (c) Multi-view relation learning. We first briefly introduce few-shot learning for completeness before diving deep into MVFSL.

2.1 Few-shot Learning

For few-shot image learning, there are two types of sets, namely, *support set* S and *query set* Q. C unique classes with K labeled images for each of C classes are randomly sampled from the training set to form $S = \{(x_i, y_i)\}_{i=1}^{m}$ $(m = K \times C)$, where x_i is one sample image and y_i is its label. The *query set* $Q = \{(x_j, y_j)\}_{j=1}^{n}$ is constructed using remaining samples of selected C classes. The support set and query set together form a training episode. Typically, K is a small number for few-shot setting, e.g., $K = 1$ or 5. The task is denoted as C-way K-shot learning. In the training, pairwise images from Q and S are constructed for model learning. In the test stage, we adopt similar strategy to construct both Q and S from the test set and classifies the images from a query set by assigning each image with a label. Note that categories from both training set and test set are disjoint. The time cost of few-shot food recognition does not depend on the number of the whole class, but depends on the value of selected K and C. Therefore, there is no direct relevance between the time cost of few-shot food recognition and the number of the whole classes for each test sample in the test phrase. Generally, image pairs from Q and S constructed from the training set can be very large, and deep learning networks thus can be utilized for few-shot learning [37, 57].

2.2 Category and Ingredient-oriented Feature Learning

For food category oriented feature learning, we use the training set with categories to fine-tune one deep network. We then extract feature maps $f_{\mu}(x_i)$ of the last convolution layer, where μ are model parameters. In addition, we use images from the training set and their associated multi-label ingredients to fine-tune another deep network for multi-label ingredient attribute learning, and extract ingredient-oriented feature maps $f_{\nu}(x_i)$, where ν denotes parameters of the ingredient-oriented deep network. In multi-label ingredient attribute learning, we consider multi-label ingredient learning with M ingredients as M binary attribute classification tasks, where M is the size of ingredient vocabulary. Note that any deep networks can be used in this stage. Without loss of generality, we adopt VGG-16 [55] as the backbone network to introduce our method.

2.3 Multi-view Feature Map Fusion

Feature maps of a deep convolutional layer are usually sparse and tend to be selective of higher-level visual concepts, as observed in References [54, 63]. To introduce our method, we first

[1] https://github.com/minweiqing/Few-Shot-Food-Recognition-via-Multi-View-Representation.

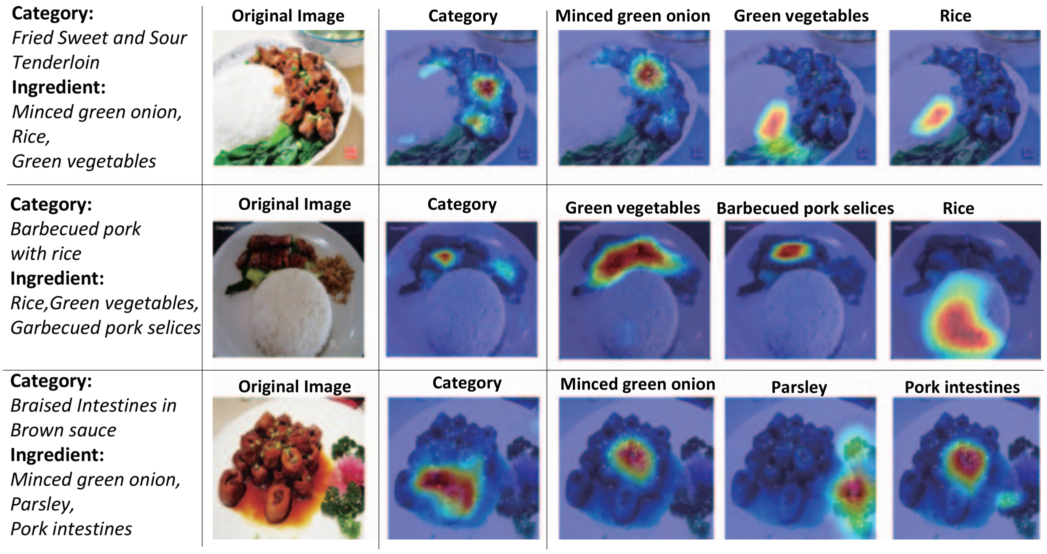


Fig. 3. Discriminative localization maps from some food images. Grad-CAM [54] is adopted to implement the category and ingredient discriminative localization region visualization(the warmer the color of the overlay image, the more discriminative that pixel is). From left to right: (1) Category and its ingredients, (2) Original images, (3) Category-discriminative localization maps, and (4) Ingredient-discriminative localization maps.

demonstrate three images from VIREO Food-172 and their visualization results in Figure 3, where the visualization is realized via Grad-CAM [54]. Note that for ingredients, we train a multi-label classification model and then obtain discriminative localization regions via gradients for each ingredient label. From Figure 3, we find that activated regions of many feature maps (highlighted in warm colors) are semantically meaningful. For example, the activated region of "Rice" tends to localize at the rice region of "Fried Sweet and Sour Tenderloin" and "Barbecued Pork with Rice." The activated region of "Parsley" tends to localize at the parsley region of "Braised Intestines in Brown sauce." Therefore, ingredient-oriented features are capable of capturing additional details, which are complementary to category-oriented features. For example, for "Braised Intestines in Brown sauce," the ingredient region covers different parts of the image, such as activated regions of "Parsley" and "Minced green onion," which are complementary with activated regions of the category.

Based on above-mentioned observation and analysis, we could combine category and ingredient activated regions for enhanced feature representation. Particularly, we calculate the combined feature map representation $\zeta(f_\mu(x_i), f_v(x_i))$ via the operator $\zeta(\cdot)$. There are many ways for feature combination. In this article, the operator $\zeta(\cdot)$ is the concatenation of feature maps in depth. For example, the extracted feature maps form the last convolution layer with 14×14×512 from two types of fine-tuned deep networks, the size of combined features will be 14×14×1024 via $\zeta(\cdot)$.

After the combination between feature maps from two different types, we then conduct the convolution on these combined feature maps. Meanwhile, the convolution is conducted with the following multi-view relation learning together for parameter learning and will give more details in the following subsection. Our adopted multi-view feature map concatenation and convolution fusion is similar to Reference [16]. However, they focus on spatiotemporal fusion from video for activation classification. In contrast, we conduct both ingredient-oriented and category-oriented fusion for few-shot food image recognition. In addition, the fused features contain richer information from food attributes, and thus bridge the gap between disjoint training categories and test categories for the performance improvement of few-shot food recognition.

2.4 Multi-view Relation Learning

Multi-view relation learning is used to compare query images against labeled sample images to determine if these images are from matching categories or not based on the image-level relation score.

For multi-view relation learning, we sample the fused multi-view representation $\zeta(f_\mu(x_i), f_\nu(x_i))$ from the support set S and $\zeta(f_\mu(x_j), f_\nu(x_j))$ from the query set Q. These feature maps $\zeta(f_\mu(x_i), f_\nu(x_i))$, and $\zeta(f_\mu(x_j), f_\nu(x_j))$ are combined via $\tau(\cdot)$, where the operator $\tau(\cdot)$ is also the concatenation of feature maps in depth. For example, the fused feature maps from both x_i and x_j are $\zeta(f_\mu(x_i), f_\nu(x_i))$, and $\zeta(f_\mu(x_j), f_\nu(x_j))$, respectively. Their size is 14×14×1,024. After concatenation $\tau(\cdot)$, the final dimension is 14×14×2,048. The combined feature maps of samples from the query and support set are further processed via the relation subnetwork h_ϕ with some convolutional blocks and full-connected layers to generate the relation score:

$$\gamma_{i,j} = h_\phi(\tau(\zeta(f_\mu(x_i), f_\nu(x_i)), \zeta(f_\mu(x_j), f_\nu(x_j)))). \tag{1}$$

The mean-square error is then used to train the model, regressing the relation score $\gamma_{i,j}$ to the ground truth: matched pairs have similarity 1 and mismatched pairs have similarity 0. The final objective is as follows:

$$\underset{\phi}{\operatorname{argmin}} \sum_{i=1}^{m} \sum_{j=1}^{n} (\gamma_{i,j} - \mathbf{1}(y_i == y_j))^2, \tag{2}$$

where m is the number of images from the support set S and n is the number of images from the query set Q.

Note that the proposed MVFSL is inspired by Relation Network (RN) [57] for few-shot learning, but with two important differences: (1) RN only learns category oriented visual features for few-shot learning, while MVFSL can further utilize ingredient-oriented features for few-shot learning. (2) RN only applies convolution to category oriented feature maps. In contrast, we utilize the convolution on fused multi-view feature maps for multi-view relation learning. Through the convolution on feature maps with two different types, category-oriented and ingredient-oriented features are effectively fused. In addition, the convolution is further conducted based on feature maps between two images for image-level relation learning.

2.5 Optimization

We introduce two settings for training MVFSL. In the first setting, we first fine-tune food category-supervised deep network and ingredient-supervised deep network, respectively. The corresponding feature maps are extracted. These two types of feature maps are then fused and the fused features are finally fed into the relation network for multi-view relation learning. Such setting is Loosely Combined and named as MVFSL-LC. In the second setting, after fine-tuning both food category-supervised deep network and ingredient-supervised deep network, the whole training on these two types of subnetworks and relation subnetworks in MVFSL are further conducted in an end-to-end fashion for joint optimization. This setting is Tightly Combined and named as MVFSL-TC.

In the test stage, we obtain combined features from the support set and query set via multi-view feature map fusion, and h_ϕ is then used to generate the relation score between query set and each of support set. Finally, we can make the prediction according to the maximum relation score.

3 BASELINE METHODS OF INGREDIENT-BASED FEW-SHOT LEARNING NETWORKS

The ingredients can also be fused to other few-shot learning methods. In this section, we introduce ingredients into another two popular networks of few-shot learning methods, namely, Siamese Network (SN) [34] and Matching Network (MN) [60], where multi-view relation learning is replaced with fixed cosine distance calculation between two images.

3.1 Siamese Network

When Siamese Network (SN) is used for few-shot learning, it first randomly samples image pairs with the same class or different classes from the training set to learn the model. In the test stage, the trained model is used to classify the unlabelled image X into one of C categories from the test set. Given an unlabeled image X and other images $\{X_c\}_{c=1}^{C}$, where X_c represents one example with the label c. The image pairs $(X, X_c)_{c=1}^{C}$ are fed into the trained model to make the prediction for X according to the cosine similarity between this image and other samples from those C categories.

To exploit ingredient information, for category oriented feature learning, we replace the original network of SN with VGG-16 for fair comparison, and then fine-tune the network under the few-shot learning setting to extract more discriminative visual features $f_\mu(x_i)$ for each test image x_i. For ingredient oriented feature learning, each food image is associated with multiple ingredients. However, SN cannot be trained for multi-label classification. To solve this problem, we first fine-tune this network for multi-label ingredient classification. We then initialize SN using this fine-tuned network, and further fine-tune the last layer of SN for few-shot food recognition. Finally, we use the fine-tuned SN to extract ingredient-oriented features $f_\nu(x_i)$ for each test image x_i. A multi-view feature fusion $\zeta(\cdot)$ is used to combine these two kinds of features $f_\mu(x_i)$ and $f_\nu(x_i)$ into a unified representation. Because we adopt fixed cosine similarity calculation, the corresponding features from the fine-tuned network are from the fc7 layer. We finally use the fused features to predict the unlabelled image x according to the cosine similarity between this image and other images from C categories.

3.2 Matching Network

Matching Network (MN) learns a network that maps a small labelled support set and an unlabeled query example to its label. In the training stage, it learns a classifier $c_S(x)$ from the support set S_{train} that constructed from the training set. The classifier can be defined as one mapping function $P_\theta(x|y, S_{train})$, where θ are parameters of the model, and should be learned in the training. In the test stage, given an unlabeled example x_t and a support set S_{test} from the test set, MN uses learned $P_\theta(.|x_t, S_{test})$ to predict its label y_t.

We adopt similar strategy to extend MN by exploiting ingredient information. We obtain fused features for each test sample under the subjective function of MN. Then cosine similarity between the query sample and each sample in the support set is calculated to make prediction.

4 EXPERIMENT

In this section, we first describe the experimental setting including the dataset and implementation details. We then evaluate the performance of MVFSL qualitatively and quantitatively on different food datasets. Next, we evaluate the performance of another two extended few-shot learning methods Siamese Network and Matching Network. Finally, we give additional analysis and discussions.

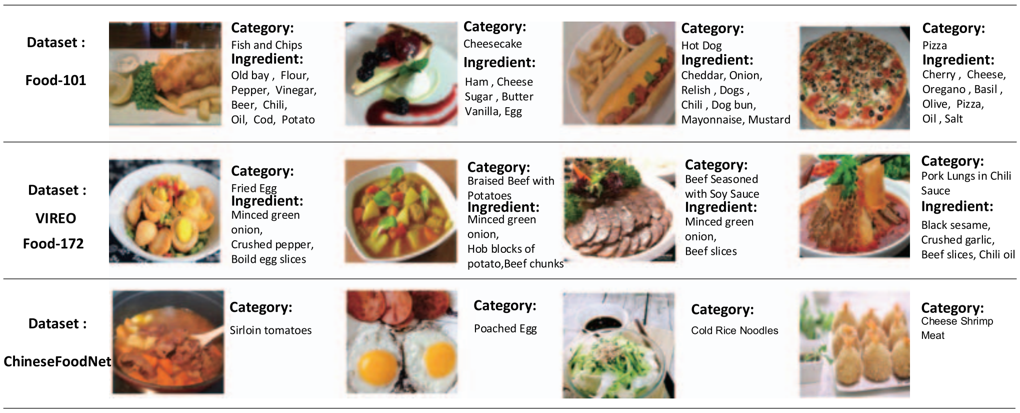

Dataset :
Food-101
Category:
Fish and Chips
Ingredient:
Old bay , Flour, Pepper, Vinegar, Beer, Chili, Oil, Cod, Potato
Category:
Cheesecake
Ingredient:
Ham , Cheese Sugar , Butter Vanilla, Egg
Category:
Hot Dog
Ingredient:
Cheddar, Onion, Relish , Dogs , Chili , Dog bun, Mayonnaise, Mustard
Category:
Pizza
Ingredient:
Cherry , Cheese, Oregano , Basil , Olive, Pizza, Oil , Salt
Dataset :
VIREO
Food-172
Category:
Fried Egg
Ingredient:
Minced green onion, Crushed pepper, Boild egg slices
Category:
Braised Beef with Potatoes
Ingredient:
Minced green onion, Hob blocks of potato,Beef chunks
Category:
Beef Seasoned with Soy Sauce
Ingredient:
Minced green onion, Beef slices
Category:
Pork Lungs in Chili Sauce
Ingredient:
Black sesame, Crushed garlic, Beef slices, Chili oil
Dataset :
ChineseFoodNet
Category:
Sirloin tomatoes
Category:
Poached Egg
Category:
Cold Rice Noodles
Category:
Cheese Shrimp Meat

Fig. 4. Some examples from Food-101, VIREO Food-172, and ChineseFoodNet.

4.1 Experimental Setting

4.1.1 Dataset. Since there is no food dataset for few shot learning, we use the following three food datasets, namely, Food-101 [9], VIREO Food-172 [11], and ChineseFoodNet [12] to simulate few-shot food recognition.

Food-101 contains 101,000 images with 101 classes in total, where most categories belong to the western food. To use the dataset to simulate few-shot food recognition, similar to References [57, 60], we randomly split Food-101 into 71 classes and 30 classes for the training set and test set, respectively. For ingredient information, we adopt the ingredients from Ingredients101 [8] with 446 ingredients in total and 9 ingredients for each image on average.

VIREO Food-172 contains 172 categories. All the images in the data set are Chinese food. Similarly, we randomly split the data set into 132, 40 classes for the training and test set. There are 353 ingredients in total with 3 ingredients for each image on average.

ChineseFoodNet covers many popular Chinese food items from different styles of cooking. This dataset contains 185,628 images with 208 Chinese dish categories. We randomly split the data set into 158 classes for training and 50 classes for testing, respectively. Both VIREO Food-172 and ChineseFoodNet belong to Chinese cuisine, and thus share lots of ingredients. Considering the ingredient information is not provided in ChineseFoodNet, we adopt the trained ingredient-supervised deep network from VIREO Food-172 as the ingredient model for ChineseFoodNet.

Figure 4 shows some samples with ingredients from three datasets, respectively. Note that many works, such as References [57, 60] conduct few shot learning in the miniImageNet dataset with 100 classes, each having 600 examples. Eighty classes are used for training and the remaining 20 classes for test. The few-shot food recognition belongs to this scenario. These food dataset is similar to this miniImageNet with similar scale and similar training-test class split, and therefore can be regarded as in real scenarios for few-shot learning.

4.1.2 Implementation Details. In MVFSL, there are three subnetworks, category-oriented subnetwork, ingredient-oriented subnetwork, and multi-view relation subnetwork. The first two types of subnetworks adopt VGG-16 network without fully-connected layers and the classification layer. Similar to Reference [57], multi-view relation subnetwork consists of two convolutional blocks and two fully-connected layers, where each of convolutional blocks is a 3 × 3 convolution with 64 filters followed by the batch normalization, the ReLU non-linearity and 2 × 2 maxpooling. The two fully-connected layers are eight- and one-dimensional, respectively.

Table 1. Performance Comparison on MVFSL

Model	Food-101	VIREO Food-172	ChineseFoodNet
RN-Category [57]	53.9%	74.0%	63.8%
RN-Ingredeint	53.5%	70.5%	64.0%
MVFSL-LC	55.1%	74.8%	65.8%
MVFSL-TC	**55.3%**	**75.1%**	**66.1%**

For our model, following existing few-shot learning setting [57], an episode-based training strategy is adopted. We randomly select $C = 5$ classes and sample one image $K = 1$ from each selected category as the support set S. That is, a common five-way one-shot setting is adopted in the training stage. For the query set Q, 15 query images are sampled from each selected category. There are $15 \times 5 + 1 \times 5 = 80$ images in each episode. In the training stage, we sample 100,000 episodes from the training set. The Adam [33] is used to perform stochastic optimization over few-shot learning with the initial learning rate 10^{-4} and reduced by half for every 20,000 episodes. The accuracy of few-shot classification is computed by averaging over 1,000 episodes randomly generated from the test set.

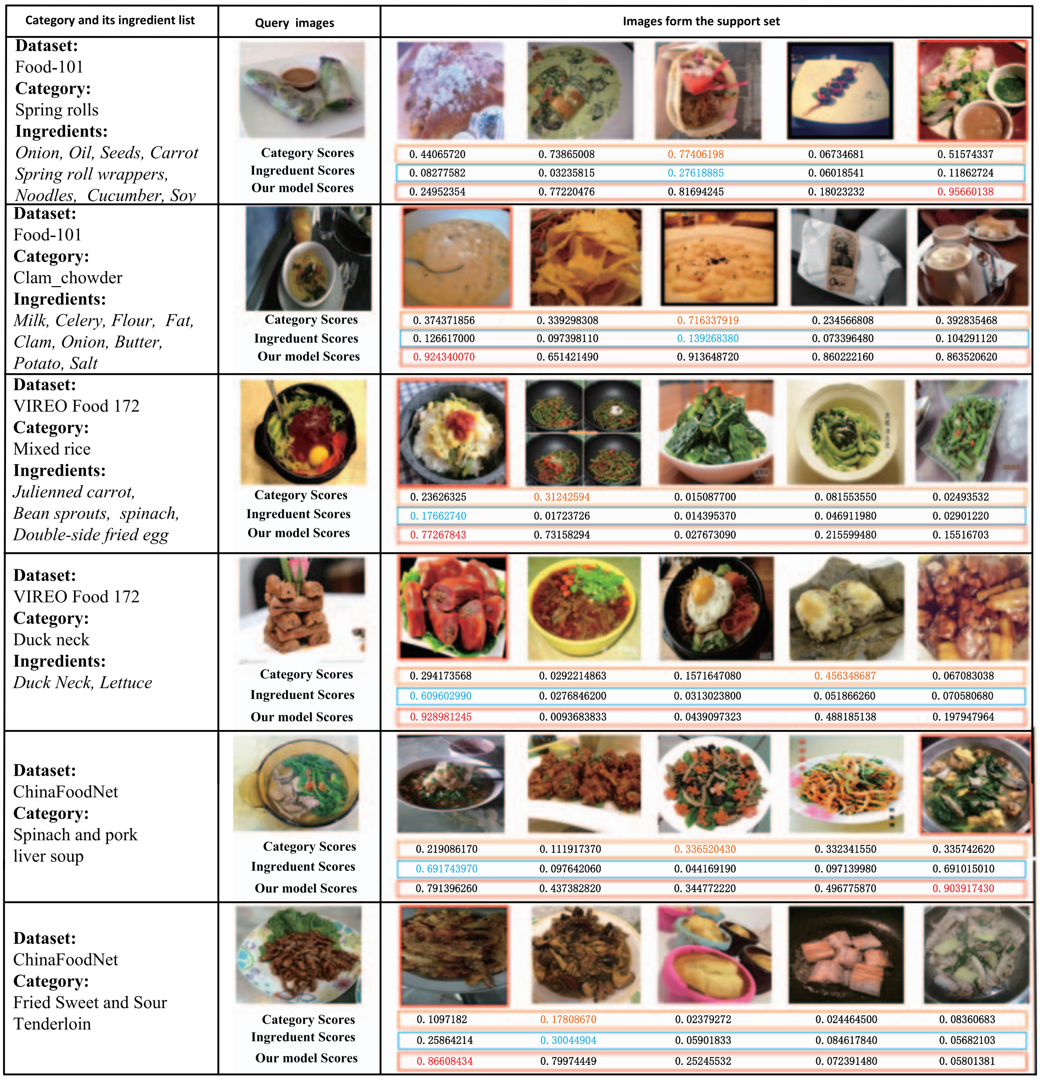
4.2 Experimental Evaluation for MVFSL

4.2.1 Quantitative Evaluation. Considering there are no methods for few-shot food recognition, we design the following baselines to demonstrate the effectiveness of MVFSL:

- Relation Network-Category (RN-Category) [57]: This baseline uses images and their categories to train the Relation Network.
- Relation Network-Ingredient (RN-Ingredient): This baseline uses the ingredient information and images to train the Relation Network.

We conduct comprehensive evaluation on three datasets, respectively. Note that we should utilize samples from VIREO Food-172 to learn ingredient-oriented model for ChineseFoodNet. In the training stage, we first remove test classes of ChineseFoodNet from the training set of VIREO Food-172 dataset, and then use the rest training images and their corresponding ingredients from VIREO Food-172 to learn ingredient-based model, which can be used to extract ingredient-oriented features for ChineseFoodNet. Table 1 summarizes the experimental results. We can see that MVFSL-LC and MVFSL-TC on three datasets perform better than their corresponding baselines. Particularly, for Food-101, MVFSL-LC achieves better performance compared with RN-Category and RN-Ingredient, and outperforms those two baselines by 1.2% and 1.6%, respectively; MVFSL-TC achieves best performance compared with MVFSL-LC. We can see similar trends for both VIREO Food-172 and ChineseFoodNet. All these experimental results validated the effectiveness of MVFSL in fusing both category-oriented and ingredient-oriented feature representations. In addition, ingredient oriented information can bridge the gap between disjoint training categories and test categories to enable the performance improvement. In addition, MVFSL-TC jointly optimizes parameters from all subnetworks and thus can achieve the best performance.

4.2.2 Qualitative Evaluation for MVFSL. We further demonstrated the effectiveness of MVFSL by showing some cases. Figure 5 shows some examples from MVFSL. We can see that (1) RN-Ingredient could make more reasonable predictions compared with RN-Category in many cases. For some cuisines, such as "Mixed rice," "Duck neck," and "Spinach and pork liver soup," RN-Category failed to make the correct prediction, while RN-Ingredient made the correct prediction. (2) MVFSL could make accurate prediction for some cuisines that are quite difficult to recognize



Category and its ingredient list	Query images	Images form the support set				
Dataset: Food-101 **Category:** Spring rolls **Ingredients:** *Onion, Oil, Seeds, Carrot Spring roll wrappers, Noodles, Cucumber, Soy*	Category Scores Ingreduent Scores Our model Scores	0. 44065720 0. 08277582 0. 24952354	0. 73865008 0. 03235815 0. 77220476	0. 77406198 0. 27618885 0. 81694245	0. 06734681 0. 06018541 0. 18023232	0. 51574337 0. 11862724 0. 95660138
Dataset: Food-101 **Category:** Clam_chowder **Ingredients:** *Milk, Celery, Flour, Fat, Clam, Onion, Butter, Potato, Salt*	Category Scores Ingreduent Scores Our model Scores	0. 374371856 0. 126617000 0. 924340070	0. 339298308 0. 097398110 0. 651421490	0. 716337919 0. 139268380 0. 913648720	0. 234566808 0. 073396480 0. 860222160	0. 392835468 0. 104291120 0. 863520620
Dataset: VIREO Food 172 **Category:** Mixed rice **Ingredients:** *Julienned carrot, Bean sprouts, spinach, Double-side fried egg*	Category Scores Ingreduent Scores Our model Scores	0. 23626325 0. 17662740 0. 77267843	0. 31242594 0. 01723726 0. 73158294	0. 015087700 0. 014395370 0. 027673090	0. 081553550 0. 046911980 0. 215599480	0. 02493532 0. 02901220 0. 15516703
Dataset: VIREO Food 172 **Category:** Duck neck **Ingredients:** *Duck Neck, Lettuce*	Category Scores Ingreduent Scores Our model Scores	0. 294173568 0. 609602990 0. 928981245	0. 0292214863 0. 0276846200 0. 0093683833	0. 1571647080 0. 0313023800 0. 0439097323	0. 456348687 0. 051866260 0. 488185138	0. 067083038 0. 070580680 0. 197947964
Dataset: ChinaFoodNet **Category:** Spinach and pork liver soup	Category Scores Ingreduent Scores Our model Scores	0. 219086170 0. 691743970 0. 791396260	0. 111917370 0. 097642060 0. 437382820	0. 336520430 0. 044169190 0. 344772220	0. 332341550 0. 097139980 0. 496775870	0. 335742620 0. 691015010 0. 903917430
Dataset: ChinaFoodNet **Category:** Fried Sweet and Sour Tenderloin	Category Scores Ingreduent Scores Our model Scores	0. 1097182 0. 25864214 0. 86608434	0. 17808670 0. 30044904 0. 79974449	0. 02379272 0. 05901833 0. 25245532	0. 024464500 0. 084617840 0. 072391480	0. 08360683 0. 05682103 0. 05801381

Fig. 5. Some experimental results from MVFSL and other two baselines. From left to right: (1) category and its ingredient list, (2) query images, and (3) images from the support set. In each example, we show the relation scores from MVFSL and other two baselines. The higher relation score, the more relevant the image is. The ground truth of food images is highlighted with red box.

from the support set, such as "Spring rolls," "Clam chowder," and "Fired Sweet and Sour Tenderloin." Both RN-Category and RN-Ingredient failed to make the correct prediction while MVFSL has made the correct prediction by fusing category-oriented and ingredient-oriented visual feature information. This further verified that category oriented features and ingredient oriented features are complementary, and MVFSL is capable of fusing these two types of information to improve the performance of few-shot food recognition.

4.2.3 Multi-view Relation Learning with Different Convolution Layers. For Relation Network, the combined feature maps are fed into the relation subnetwork to obtain the relation score. In this

Table 2. The Performance of MVFSL-LC and MVFSL-TC with Different Relation Network Settings on Food-101

Model	64 filters	128 filters
MVFSL-LC (1 Conv layer)	54.7%	55.9%
MVFSL-TC (1 Conv layer)	55.1%	56.0%
MVFSL-LC (2 Conv layers)	55.1%	56.3%
MVFSL-TC (2 Conv layers)	55.3%	56.4%
MVFSL-LC (3 Conv layers)	56.1%	56.9%
MVFSL-TC (3 Conv layers)	56.6%	**57.6%**
MVFSL-LC (4 Conv layers)	54.6%	55.6%
MVFSL-TC (4 Conv layers)	55.7%	56.3%

Table 3. The Performance with Different Networks for Feature Learning on Food-101, VIREO Food-172, and ChineseFoodNet

Model	Food-101			VIREO Food-172			ChineseFoodNet		
	AlexNet	VGG16	VGG19	AlexNet	VGG16	VGG19	AlexNet	VGG16	VGG19
RN-Category [57]	48.6%	53.9%	54.7%	68.8%	74.0%	74.6%	59.5%	63.8%	62.9%
RN-Ingredient	51.4%	53.5%	55.1%	70.3%	70.5%	73.6%	59.4%	64.0%	65.6%
MVFSL-LC	51.8%	55.1%	55.9%	70.8%	74.8%	75.2%	59.9%	65.8%	66.5%
MVFSL-TC	**52.1%**	**55.3%**	**56.5%**	**71.0%**	**75.1%**	**75.3%**	**60.2%**	**66.1%**	**66.7%**

experiment, we conducted the performance analysis when varying the number of convolution blocks and filters. Table 2 shows experimental results on Food-101 from MVFLS-LC and MVFLS-TC. We can see that: (1) When we fix the number of filters, with the increase of the number of convolution layers, there is a consistent increase in the performance. (2) The network with 128 filters achieves better performance than the one with 64 filters. For example, in MVFLS-LC, with the increase of the number of convolution layers, the network with 128 filters achieves better performance than the network with 64 filters and outperforms it by 0.2%, 1.2%, 0.8%, and 1.0% for different convolutional layers, respectively. (3) At every group of parameter setting, MVFLS-TC achieves better performance than MVFLS-LC. (4) MVFSL achieved the best performance at the setting of 3 convolution layers and 128 filters, With the increase of the number of layers and filters, the performance of MVFSL reduces. This is because the increased complexity of the model probably leads to overfitting.

4.2.4 The Depth of Basic Network for MVFSL. We show that the performance of our method with different layers of deep networks. As shown in Table 3 on Food-101, we can see that: (1) With the increase of layers, there is consistent increase in the performance. For example, our method and baselines with the VGG16 network achieve better performances than AlexNet, and outperform AlexNet-based RN-Category, RN-Ingredient, MVFSL-LC, and MVFSL-TC by 5.3%, 2.1%, 3.3%, and 3.2%, respectively. The reason is that the deeper network can extract more discriminative features, which are helpful for few-shot food recognition. (2) The performance of MVFSL is always better than other baselines for the same deep architecture. Particularly, for the AlexNet network, the performance of MVFSL-LC outperforms RN-Category and RN-Ingredient by 3.2% and 0.4%. For the VGG16 Network, there is also the performance improvement compared with RN-Category and RN-Ingredient, and outperforms those two models by 2.1% and 1.6%, respectively. We can see similar performance improvement for the model with VGG19 Network. (3) MVFSL-TC with different layers of deep networks achieves consistent increase in the performance compared with MVFSL-LC. For example, for VGG19-based MVFSL, MVFSL-TC outperforms MVFSL-LC by 0.6%.

Table 3 also provides comparative results on VIREO Food-172 and ChineseFoodNet. Though there are different categories between ChineseFoodNet and VIREO Food-172, both of them are Chinese food and their ingredients are similar, and we thus could still obtain the ingredient representation of ChineseFoodNet using the trained ingredient model. We can observe that: (1) With the increase of layers, MVFSL and other baselines similarly obtain consistent performance increase. The MVFSL and other baselines with the VGG16 network outperform those models with AlexNet by 0.2% to 5.2%. Similarly, in most cases, the performance of MVFSL and other baselines with VGG19 network is better than those model with VGG16 network. There is only one exception that RN-Category with VGG19 achieves lower performance compared with VGG16. One probable reason is that there is overfitting in the network training on this dataset. (2) There is consistent increase for MVFSL compared with the baselines for each architecture with the same layers. (3) For MVFSL-TC with AlexNet, VGG16, and VGG19, there is consistent performance increase compared with MVFSL-LC. This again verified the effective of end-to-end training.

4.3 Experimental Evaluation for SN and MN

For the evaluation setting of SN, MN, and RN, we conduct the comparison according to Reference [57] for fair evaluation. For SN, we need to construct positive image pairs and negative image pairs, which should conform the setting of five-way one-shot with one query. Participially, we randomly select five categories and sample one image from each selected category as the support set. One query image is sampled from one of those selected categories. We construct the positive image pair and negative image pair by combining the query image with each image from the support set, where the positive image pair means these images are from the same class and negative image pair means they are from different classes. There are four negative pairs and one positive pair in each five-way one-shot with one query setting. We randomly sample 50,000 image pairs with 10,000 positive pairs and 40,000 negative pairs in each training epoch. The classification accuracy is computed by averaging over 5,000 randomly generated image pairs from the test set. Under the five-way one-shot with one query setting, 5,000 randomly generated image pairs are equivalent to 1,000 tasks. Nesterow Momentum is used to perform stochastic optimization with the initial learning rate 5×10^{-5} and momentum value 0.9.

For MN, we randomly select five categories and sample one image from each selected category as the support set, and also one query image from one of those selected categories as the query set, which means there are $5 \times 1 + 1 = 6$ images in each episode. In the training stage, we sample 100,000 episodes from the training set. Adam [33] is used for stochastic optimization with the initial learning rate 5×10^{-4} and half the learning rate for every 20,000 episodes. The few-shot classification accuracy is computed by averaging over 1,000 randomly generated episodes from the test set.

To further demonstrate the effectiveness in introducing the ingredient information, we use the following baselines to evaluate SN-Multiview and MN-Multiview:

- Siamese Network-Category (SN-Category) [34]: This baseline uses images and their categories to train the Siamese Network.
- Siamese Network-Ingredient (SN-Ingredient): This baseline uses images and their ingredients to train the Siamese Network.
- Matching Network-Category (MN-Category) [60]: This baseline uses images and their categories to train the Matching Network.
- Matching Network-Ingredient (MN-Ingredient): This baseline uses images and their ingredients to train the Matching Network.

Table 4. Performance Comparison on SN

Model	Food-101	VIREO Food-172	ChineseFoodNet
SN-Category [34]	49.1%	60.3%	50.5%
SN-Ingredient	54.5%	65.5%	62.5%
SN-Multiview	**55.0%**	**65.8%**	**64.4%**

Table 5. Performance Comparison on MN

Model	Food-101	VIREO Food-172	ChineseFoodNet
MN-Category [60]	45.6%	73.6%	48.9%
MN-Ingredeint	46.8%	65.9%	52.0%
MN-Multiview	**47.5%**	**74.1%**	**53.0%**

Table 6. Performance Comparison on Different Methods

Model	Food-101	VIREO Food-172	ChineseFoodNet
SN-Multiview	55.0%	65.8%	64.4%
MN-Multiview	47.5%	74.1%	53.0%
MVFSL-LC	55.1%	74.8%	66.5%
MVFSL-TC	**55.3%**	**75.1%**	**66.7%**

Tables 4 and 5 provide the comparative results. We can observe that there is a consistent increase for both SN-Multiview and MN-Multiview compared with their corresponding baselines. Particularly, for SN-Multiview, we can achieve performance improvement compared with the baselines and outperforms them by 4% to 5%. Similarly, MN-Multiview also achieves performance improvement compared with the baselines, and outperform them by 0.4% to 8.2%. This further verified the effectiveness of our proposed method in exploiting ingredient information.

Table 6 further summarized experimental results among MVFSL-TC, MVFSL-LC, SN-Multiview, and MN-Multiview. We can see that (1) The performance of SN-Multiview is better than MN-Multiview in most cases. This trend is consistent with experiment results on other datasets [57]. (2) The performance of MVFSL-TC is better than MVFSL-LC. The reason is that MVFSL-TC is capable of making deep feature learning and relation learning reinforce each other. (3) MVFSL-TC achieves the best performance compared with other three methods. For Food-101, MVFSL-TC achieves the best performance compared with MVFSL-LC, SN-Multiview and MN-Multiview, and outperforms them by 0.2%, 0.3%, and 7.8%. Similarly, MVFSL-TC achieves the best performance for VIREO Food-172 and ChineseFoodNet.

4.4 Experimental Evaluation for Different C-way K-shot

In this section, we conducted the comparison under different C-way K-shot settings. Table 7 provides experimental results on three datasets. We can observe that (1) there is consistent increase for MVFSL-LC and MVFSL-TC on different C-way K-shot setting compared with RN-Category and RN-Ingredient. Furthermore, there is marginal improvement for end-to-end training and MVFSL-TC achieves best performance. For example, on the Food-101, for the five-way one-shot setting, MVFSL-LC achieves better performance compared with RN-Category and RN-Ingredient, and outperforms those two baselines by 1.2% and 1.6%, respectively, and MVFSL-TC achieves better performance compared with MVFSL-LC. Also similar trends for other C-way K-shot settings.

Table 7. The Performance on Different Networks with Different C-way K-shot Settings on Three Datasets

Model	Food-101			VIREO Food-172			ChineseFoodNet		
	5-way 1-shot	5-way 5-shot	20-way 1-shot	5-way 1-shot	5-way 5-shot	20-way 1-shot	5-way 1-shot	5-way 5-shot	20-way 1-shot
RN-Category [57]	53.9%	67.4%	25.5%	74.0%	83.0%	47.5%	63.8%	72.9%	34.1%
RN-Ingredient	53.5%	67.5%	25.8%	70.5%	76.4%	40.5%	64.0%	73.0%	34.3%
MVFSL-LC	55.1%	68.1%	26.5%	74.8%	83.5%	48.2%	65.8%	73.5%	34.4%
MVFSL-TC	**55.3%**	**68.3%**	**26.8%**	**75.1%**	**83.6%**	**48.6%**	**66.1%**	**73.9%**	**34.6%**

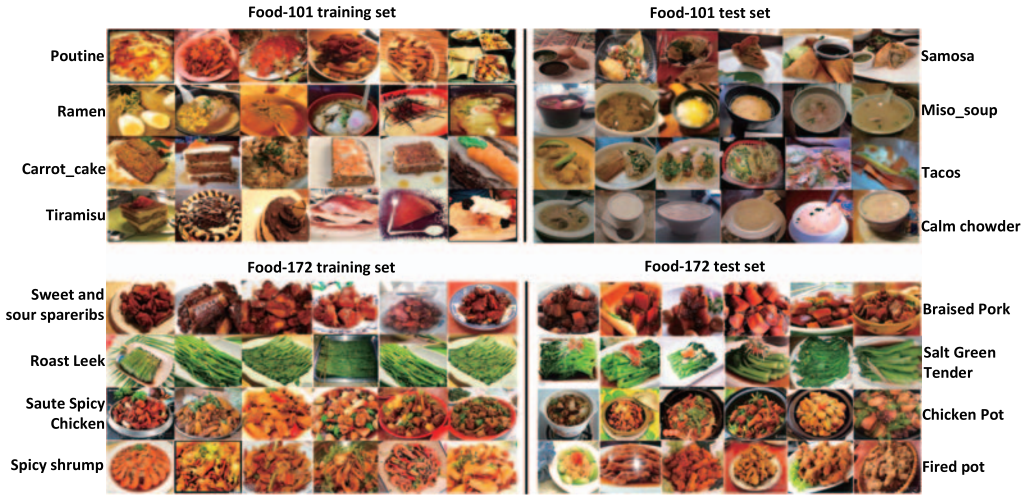
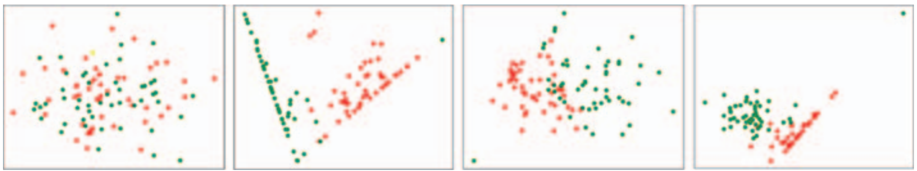


Fig. 6. Examples of Food-101 and VIREO Food-172.

4.5 Discussions

4.5.1 What Causes the Different Performances? We notice that the performance on both VIREO Food-172 and ChineseFoodNet are better than the Food-101 dataset. The probable reason lies in the difference between the training set and test set. In the few-shot learning, the classes from the training set and test set are disjoint. Although introducing the ingredient information can release this problem, if there is larger difference between the distribution of the training set and the test data, then the method only learns less information for the test set in the training stage. Conversely, if there are more similar distributions between them, such as the appearance and shape, then the performance can increase. Figure 6 shows some examples selected from Food-101 and VIREO Food-172. For Food-101, the appearance of the training set are quiet different from test set. In contrast, for VIREO Food-172, there is a more similarity in some aspects (e.g.,the appearance and color) for some categories, such as "Roast Leek" from the training set and "Salt Green Tender" from the test set, and "Fired Sweet and Sour Tenderloin" of the training set and "Braised Pork" of the test set. Such similar distribution between training set and test set enables transfer learning between these two sets for performance improvement.

4.5.2 Visualization. In this section, we qualitatively analyze the results of relation learning. We randomly sample two categories with 100 images from the Food-101 dataset and all the images are projected to 2D by PCA [57]. Figure 7(a) shows real sample images colored by matching (red) or mismatching (green) images and query image (yellow). We can see that comparing embeddings of original images are very challenging. In Figures 7(b)–7(d), we then qualitatively show the relation representations of RN-Category, RN-Ingredient and MVFSL-LC-based matched (red)



(a) Matched (red) and mismatched (green) sample embeddings for a given query (yellow)

(b) RN-Category: Matched (red) and mismatched (green) relation module pair representations

(c) RN-Ingredient: Matched (red) and mismatched (green) relation module pair representations

(d) MVFSL-LC:Matched (red) and mismatched (green) relation module pair representations

Fig. 7. Examples of Food-101 few-shot problem visualizations. (a) Matched (red) and mismatched (green) sample embedding for a given query (yellow); (b) RN-Category: Matched (red) and mismatched (green) relation module pair representations; (c) RN-Ingredient: Matched (red) and mismatched (green) relation module pair representations; and (d) MVFSL-LC: Matched (red) and mismatched (green) relation module pair representations.

and mismatched (green) query-sample pairs, respectively. Similar to Reference [57], we plot each query-sample pair that represented by relation module pair representations. We can see that relation network has mapped query-sample pairs into a linearly separable space. In addition, MVFSL-LC-based (mis)matched query-sample pairs are more linearly separable.

5 RELATED WORK

Our work is closely related to the following two research areas: (1) food image classification and (2) few-shot learning.

5.1 Food Image Recognition

Recently, Min et al. [45] provided a comprehensive survey of food recognition and other food-related works. Food recognition is a difficult problem, since foods can dramatically vary in appearance. Such variations may arise not only from changes in illumination and viewpoint but also from non-rigid deformations, and intra-class variability in shape, texture, color, and other visual properties. Existing food recognition researches can be categorized into two major directions: (1) conventional approaches and (2) deep-learning approaches. For conventional approaches, Yang et al. [62] exploited the spatial relationship among different ingredients. The food items are represented by pairwise statistics between local features of the different ingredients of the food items. This approach is bound to work only for standardized meals. Bossard et al. [9] used the random forest method to mine discriminative parts of food images for recognition. Except the such works, plenty of researches has been carried out to find the optimal hand-crafted representation for food recognition. Joutou et al. [29] exploited several kinds of image features together with a multiple kernel learning. They combined Bag-of-SIFT with color histograms and Gabor filters to discriminate between images. Martinel et al. [41] proposed a complex scheme that can independently classify each feature response through an extreme learning machine, and combine the classification results by using a structured SVM. Kawano et al. [32] developed a real-time mobile food recognition system that it performs HoG and color patch feature encoding via fisher vector.

Compared with the conventional approaches, many deep-learning-based methods have been developed for food recognition [14, 30, 31]. For example, Kawano et al. [31] found that deep features performed significantly better than hand-crafted features. Kagaya et al. [30] extracted the deep visual features for food detection and recognition. These approaches only used deep visual features, but ignored the context. Some works [5, 7, 28, 47, 61] developed context-based recognition by introducing additional information, such as GPS, restaurant menus and ingredients. For example, Xu et al. [61] explored the geolocation and external information about restaurants to simplify the classification problem. Those works can confirm that the contextual knowledge is crucial to improve recognition. Recently, Min et al. [47] utilized ingredients as supervised signals to localize multiple regions with different scales and fused these regional features into the unified feature representation for recognition. There are also some works [49, 50], which developed mobile food recognition systems for dietary management. In addition, our work is relevant to ingredient-recipe correlation learning and cross-modal retrieval [11, 39, 44, 46]. In contrast, we take food attributes into account for few-shot food recognition.

5.2 Few-shot Learning

Few-shot learning has received more and more attention for recognizing novel visual categories from very few labeled examples. The seminal work [15] proposed a variational Bayesian framework for few-shot learning, which utilized previous learned classes to predict new ones when only one or very few examples are available. A hierarchical Bayesian program learning method [36] was later proposed to match the human level error on the few-shot alphabet recognition task. Recent works have adopted different strategies to deal with the few-shot problem [6, 17, 34, 48, 53, 56, 57, 60], which can be summarized into two main kinds of approaches. The first one is meta learning [3, 59], which tries to extract some transformable information to avoid overfitting in the few-shot learning stage. Different mechanisms, such as attention mechanism [19], memory mechanism [10] and category-agnostic activations-parameters mapping learning mechanism [51], are introduced to improve the performance of few-shot learning. The second type is metric-learning-based methods [34, 57, 60], which aim to learn a set of projection functions such that when represented in this embedding, images are easy to recognize using simple nearest neighbor or linear classifiers. In addition, some works [48, 53] leverage recurrent neural networks with memories to solve the few-shot learning problem.

Recently, there are metric-learning-based methods for few-shot learning [34, 57, 60]. For example, Koch et al. [34] employed the Siamese Networks as the embedding networks, and focused on learning the embedding to transform the data such that it can be recognized with a fixed nearest-neighbor. Later, Vinyals et al. [60] proposed the Matching Network, which transformed the support set and query samples into a shared embedding space such that it can be recognized with a fixed classifier. More recently, Sung et al. [57] proposed a model called Relation Network, which uses convolutional neural networks as a nonlinear classifier, and need not manually choose the metric, such as the cosine or Euclidean distance [19] to adapt the model or data. Furthermore, through learning a nonlinear similarity metric jointly with the embedding, this model achieves a great performance on *mini*ImageNet [60] and Omniglot [35]. Our work belongs to metric-learning-based method for few-shot learning. However, all of those works focus on using category information to solve the few-shot learning problem. In this article, we consider the problem of few-shot learning for food recognition and enhance the few-shot food recognition by leveraging ingredient attributes. In addition, few-shot learning is relevant to weakly supervised learning with small subset of training data, such as object detection and classification in optical remote sensing images [20, 21, 64]. This is because they should cope with the problem of very small subset of training data. Note that our proposed multi-view learning method focuses on learning different types of features from one image, not multiple images rendered from a 3D shape [22, 23, 24].

6 CONCLUSIONS

In this article, we have proposed a Multi-view Few-shot Learning (MVFSL) framework to explore ingredient information for few-shot food recognition. To take advantage of ingredient information, these two kinds of features are effective by first combining their extracted feature maps from the last convolution layer of their respective fine-tuned deep networks and then conducting the convolution on the combined feature maps. In addition, this convolution is incorporated into a multi-view relation network, which is used to compare query images against labeled samples to obtain the image-level relation score. MVFSL can be trained in an end-to-end way to enable joint optimization. The comprehensive experimental evaluations on three different food datasets have validated the effectiveness of MVFSL. In addition, we have extended another two types of few-shot learning methods, namely, Siamese Network and Matching Network by introducing ingredient information. The experimental results on these food datasets have also demonstrated the advantage in utilizing food ingredients for few-shot food recognition.

There are a number of issues for further study: (1) Exploring more information from food dataset to improve the performance of few-shot food recognition. For example, besides ingredient information, cooking instructions [52] and other types of attribute information, such as regional attributes and cuisine types [44] can also be utilized. (2) In our work, we have found that the difference between the distribution of training set and test set affects the performance of few-shot learning. Therefore, how to deal with this difference is worth studying. (3) We plan to use other types of advanced deep architectures such as ResNet [25] and Densenet [27] in our framework to continue improving the performance. In addition, different feature fusion strategies can also be explored, such as summation and max pooling.

REFERENCES

[1] Kiyoharu Aizawa, Yuto Maruyama, He Li, and Chamin Morikawa. 2013. Food balance estimation by using personal dietary tendencies in a multimedia food log. *IEEE Trans. Multimedia* 15, 8 (2013), 2176–2185.

[2] Giuseppe Amato, Paolo Bolettieri, Monteiro De Lira Vinicius, Cristina Ioana Muntean, Raffaele Perego, and Chiara Renso. 2017. Social media image recognition for food trend analysis. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval.* 1333–1336.

[3] Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W. Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. 2016. Learning to learn by gradient descent by gradient descent. In *Advances in Neural Information Processing Systems.* MIT Press, 3981–3989.

[4] Shuang Ao and Charles X. Ling. 2015. Adapting new categories for food recognition with deep representation. In *Proceedings of the IEEE International Conference on Data Mining Workshop.* 1196–1203.

[5] Oscar Beijbom, Neel Joshi, Dan Morris, Scott Saponas, and Siddharth Khullar. 2015. Menu-match: Restaurant-specific food logging from images. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision.* 844–851.

[6] Luca Bertinetto, João F. Henriques, Jack Valmadre, Philip Torr, and Andrea Vedaldi. 2016. Learning feed-forward one-shot learners. In *Advances in Neural Information Processing Systems.* MIT Press, 523–531.

[7] Vinay Bettadapura, Edison Thomaz, Aman Parnami, Gregory D. Abowd, and Irfan Essa. 2015. Leveraging context to support automated food recognition in restaurants. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision.* 580–587.

[8] Marc Bolaños, Aina Ferrà, and Petia Radeva. 2017. Food ingredients recognition through multi-label learning. In *Proceedings of the International Conference on Image Analysis and Processing.* Springer, 394–402.

[9] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101–mining discriminative components with random forests. In *Proceedings of the European Conference on Computer Vision.* 446–461.

[10] Qi Cai, Yingwei Pan, Ting Yao, Chenggang Yan, and Tao Mei. 2018. Memory matching networks for one-shot image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 4080–4088.

[11] Jingjing Chen and Chong-Wah Ngo. 2016. Deep-based ingredient recognition for cooking recipe retrieval. In *Proceedings of the ACM International Conference on Multimedia.* 32–41.

[12] Xin Chen, Hua Zhou, Yu Zhu, and Liang Diao. 2017. ChineseFoodNet: A large-scale image dataset for Chinese food recognition. *arXiv preprint arXiv:1705.02743.*

[13] Joachim Dehais, Marios Anthimopoulos, Sergey Shevchik, and Stavroula Mougiakakou. 2017. Two-view 3D reconstruction for food volume estimation. *IEEE Trans. Multimedia* 19, 5 (2017), 1090–1099.

[14] 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 *Proceedings of the 27th ACM International Conference on Multimedia (MM'19).* 112–120.

[15] L. Fei-Fei, R. Fergus, and P. Perona. 2006. One-shot learning of object categories. *IEEE Trans. Pattern Anal. Mach. Intell.* 28, 4 (2006), 594–611.

[16] C. Feichtenhofer, A. Pinz, and A. Zisserman. 2016. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 1933–1941.

[17] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the International Conference on Machine Learning*. 1126–1135.

[18] Jianlong Fu, Heliang Zheng, and Tao Mei. 2017. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4476–4484.

[19] Spyros Gidaris and Nikos Komodakis. 2018. Dynamic few-shot visual learning without forgetting. In *IEEE Conference on Computer Vision and Pattern Recognition*. 4367–4375.

[20] Cheng Gong, Peicheng Zhou, and Junwei Han. 2016. Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images. *IEEE Trans. Geosci. Remote Sens.* 54, 12 (2016), 7405–7415.

[21] Junwei Han, Dingwen Zhang, Gong Cheng, Lei Guo, and Jinchang Ren. 2015. Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning. *IEEE Trans. Geosci. Remote Sens.* 53, 6 (2015), 3325–3337.

[22] Zhizhong Han, Xinhai Liu, Yu-Shen Liu, and Matthias Zwicker. 2019. Parts4Feature: Learning 3D global features from generally semantic parts in multiple views. In *Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI'19)*. 766–773.

[23] Zhizhong Han, Honglei Lu, Zhenbao Liu, Chi-Man Vong, Yu-Shen Liu, Matthias Zwicker, Junwei Han, and C. L. Philip Chen. 2019. 3D2SeqViews: Aggregating sequential views for 3D global feature learning by CNN with hierarchical attention aggregation. *IEEE Trans. Image Process.* 28, 8 (2019), 3986–3999.

[24] Zhizhong Han, Mingyang Shang, Zhenbao Liu, Chi-Man Vong, Yu-Shen Liu, Matthias Zwicker, Junwei Han, and C. L. Philip Chen. 2019. SeqViews2SeqLabels: Learning 3D global features via aggregating sequential views by RNN with attention. *IEEE Trans. Image Process.* 28, 2 (2019), 1941–0042.

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 770–778.

[26] Luis Herranz, Shuqiang Jiang, and Ruihan Xu. 2017. Modeling restaurant context for food recognition. *IEEE Trans. Multimedia* 19, 2 (2017), 430–440.

[27] G. Huang, Z. Liu, L. v. d. Maaten, and K. Q. Weinberger. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2261–2269.

[28] Shuqiang Jiang, Weiqing Min, Linhu Liu, and Zhengdong Luo. 2019. Multi-scale multi-view deep feature aggregation for food recognition. *IEEE Trans. Image Process.* 29, 1 (2019), 265–276.

[29] Taichi Joutou and Keiji Yanai. 2010. A food image recognition system with multiple kernel learning. In *Proceedings of the IEEE International Conference on Image Processing*. 285–288.

[30] Hokuto Kagaya, Kiyoharu Aizawa, and Makoto Ogawa. 2014. Food detection and recognition using convolutional neural network. In *Proceedings of the ACM International Conference on Multimedia*. 1085–1088.

[31] Yoshiyuki Kawano and Keiji Yanai. 2014. Food image recognition with deep convolutional features. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. 589–593.

[32] Yoshiyuki Kawano and Keiji Yanai. 2014. Foodcam: A real-time mobile food recognition system employing fisher vector. In *Proceedings of the International Conference on Multimedia Modeling*. Springer, 369–373.

[33] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980.*

[34] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. 2015. Siamese neural networks for one-shot image recognition. In *Proceedings of the International Conference on Machine Learning*, Vol. 2.

[35] Brenden Lake, Ruslan Salakhutdinov, Jason Gross, and Joshua Tenenbaum. 2011. One shot learning of simple visual concepts. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, Vol. 33.

[36] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. 2013. One-shot learning by inverting a compositional causal process. In *Proceedings of the International Conference on Neural Information Processing Systems*. 2526–2534.

[37] Yanbin Liu, Juho Lee, Minseop Park, Saehoon Kim, Eunho Yang, Sungju Hwang, and Yi Yang. 2019. Learning to propagate labels: Transductive propagation network for few-shot learning. In *Proceedings of the International Conference on Learning Representations.*

[38] Yuzhen Lu, Yuping Huang, and Renfu Lu. 2017. Innovative hyperspectral imaging-based techniques for quality evaluation of fruits and vegetables: A review. *Appl. Sci.* 7, 2 (2017), 189.

[39] J. Marin, A. Biswas, F. Ofli, N. Hynes, A. Salvador, Y. Aytar, I. Weber, and A. Torralba. 2019. Recipe1M+: A dataset for learning cross-modal embeddings for cooking recipes and food images. *IEEE Trans. Pattern Anal. Mach. Intell.* (2019), 1. Early Access.

[40] Niki Martinel, Gian Luca Foresti, and Christian Micheloni. 2018. Wide-slice residual networks for food recognition. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*. 567–576.

[41] Niki Martinel, Claudio Piciarelli, Christian Micheloni, and Gian Luca Foresti. 2015. A structured committee for food recognition. In *Proceedings of the IEEE International Conference on Computer Vision Workshop*. 484–492.

[42] A. E. Mesas, M. Mu±ozpareja, E. Lopez-Garcia, and F. Rodríguez Artalejo. 2012. Selected eating behaviours and excess body weight: A systematic review.*Obesity Rev. Offic. J. Int. Assoc. Study Obesity* 13, 2 (2012), 106.

[43] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1233–1241.

[44] Weiqing Min, Bing-Kun Bao, Shuhuan Mei, Yaohui Zhu, Yong Rui, and Shuqiang Jiang. 2018. You are what you eat: Exploring rich recipe information for cross-region food analysis. *IEEE Trans. Multimedia* 20, 4 (2018), 950–964.

[45] Weiqing Min, Shuqiang Jiang, Linhu Liu, Yong Rui, and Ramesh Jain. 2019. A survey on food computing. *ACM Comput. Surv.* 52, 5 (2019), 92:1–92:36.

[46] Weiqing Min, Shuqiang Jiang, Jitao Sang, Huayang Wang, Xinda Liu, and Luis Herranz. 2017. Being a supercook: Joint food attributes and multimodal content modeling for recipe retrieval and exploration. *IEEE Trans. Multimedia* 19, 5 (2017), 1100–1113.

[47] Weiqing Min, Linhu Liu, Zhengdong Luo, and Shuqiang Jiang. 2019. Ingredient-guided cascaded multi-attention network for food recognition. In *Proceedings of the ACM International Conference on Multimedia*. 99–107.

[48] Tsendsuren Munkhdalai and Hong Yu. 2017. Meta networks. *arXiv preprint arXiv:1703.00837*.

[49] Kaoru Ota, Minh Son Dao, Vasileios Mezaris, and Francesco G. B. De Natale. 2017. Deep learning for mobile multimedia: A survey. *ACM Trans. Multimedia Comput. Commun. Appl.* 13, 3s (2017), 34:1–34:22.

[50] Parisa Pouladzadeh and Shervin Shirmohammadi. 2017. Mobile multi-food recognition using deep learning. *ACM Trans. Multimedia Comput. Commun. Appl.* 13, 3s (2017).

[51] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan L. Yuille. 2018. Few-shot image recognition by predicting parameters from activations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7229–7238.

[52] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3068–3076.

[53] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy P. Lillicrap. 2016. One-shot learning with memory-augmented neural networks. *CoRR* abs/1605.06065.

[54] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision Workshop*. 618–626.

[55] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[56] Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*. MIT Press, 4080–4090.

[57] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H. S. Torr, and Timothy M. Hospedales. 2018. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1199–1208.

[58] Ryosuke Tanno, Koichi Okamoto, and Keiji Yanai. 2016. DeepFoodCam: A DCNN-based real-time mobile food recognition system. In *Proceedings of the International Workshop on Multimedia Assisted Dietary Management*. 89–89.

[59] Sebastian Thrun. 1998. *Lifelong Learning Algorithms*. Springer, 181–209.

[60] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. 2016. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*. 3630–3638.

[61] Ruihan Xu, Luis Herranz, Shuqiang Jiang, Shuang Wang, Xinhang Song, and Ramesh Jain. 2015. Geolocalized modeling for dish recognition. *IEEE Trans. Multimedia* 17, 8 (2015), 1187–1199.

[62] Shulin Yang, Mei Chen, Dean Pomerleau, and Rahul Sukthankar. 2010. Food recognition using statistics of pairwise local features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2249–2256.

[63] Matthew D. Zeiler and Rob Fergus. 2013. Visualizing and understanding convolutional networks. *CoRR* abs/1311.2901.

[64] Dingwen Zhang, Deyu Meng, and Junwei Han. 2017. Co-saliency detection via a self-paced multiple-instance learning framework. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 5 (2017), 865–878.

[65] Ning Zhang, Jeff Donahue, Ross Girshick, and Trevor Darrell. 2014. Part-based R-CNNs for fine-grained category detection. In *Proceedings of the European Conference on Computer Vision*. 834–849.

[66] Jiannan Zheng, Z. Jane Wang, and Chunsheng Zhu. 2017. Food image recognition via superpixel-based low-level and mid-level distance coding for smart home applications. *Sustainability* 9, 5 (2017), 856.

Received July 2019; revised January 2020; accepted March 2020