Learning from Web Recipe-image Pairs for Food Recognition: Problem, Baselines and Performance

Bin Zhu, Chong-Wah Ngo and Wing-Kwong Chan

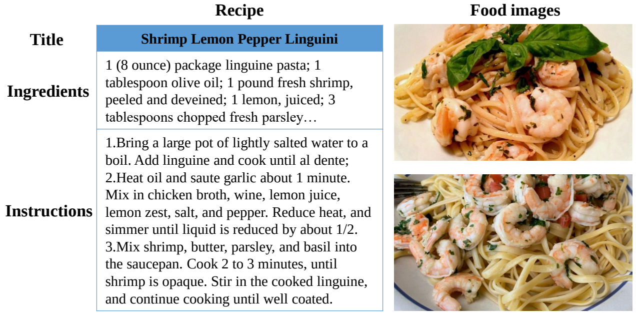
Abstract—**Cross-modal recipe retrieval has recently been explored for food recognition and understanding. Text-rich recipe provides not only visual content information (e.g., ingredients, dish presentation) but also procedure of food preparation (cutting and cooking styles). The paired data is leveraged to train deep models to retrieve recipes for food images. Most recipes on the Web include sample pictures as the references. The paired multimedia data is not noise-free, due to errors such as pairing of images containing partially prepared dishes with recipes. The content of recipes and food images are not always consistent due to free-style writing and preparation of food in different environments. As a consequence, the effectiveness of learning cross-modal deep models from such noisy web data is questionable. This paper conducts an empirical study to provide insights whether the features learnt with noisy pair data are resilient and could capture the modality correspondence between visual and text.**

Index Terms—**Food recognition, image-to-recipe retrieval, image-to-image retrieval.**

I. Introduction

Food computing is an emerging area attracting numerous research attentions recently [1]. As the famous quote “You are what you eat” [2], food profoundly influences every aspect of life from health to culture and social status [3], [4], [5]. Being able to quantify food intake, undoubtedly, is a key step towards logging the lifestyle of a person for health trend prediction. There are various factors that reflect lifestyle behaviour: food, activity, sleep pattern, emotion, gene, environment [6]. Among them, tracking food consumption is regarded as a hard problem, due to absence of physical sensors that can reliably measure nutrition consumption.

Food recognition is the fundamental building block towards the holy grail of nutrition estimation. By recognizing the category of a dish, the corresponding nutrients of the dish can be retrieved from food composition table (FCT) compiled by experts [1]. A straightforward solution is to train a recognition model in supervised learning manner by leveraging clean human annotations, such as category [7], [8], [9], ingredient composition [9], [10] as well as cutting and cooking methods [11]. Nevertheless, as the number of food categories is huge, developing classifier requires large number of training examples especially in the deep learning [12] era. It is not only tedious but also cost-expensive to label thousands or even millions of samples to train the data-hungry deep models [7], [8], [9]. The problem of image-to-recipe retrieval [13], [14] comes into picture because the category of a dish can be simply extracted from the title of a retrieved recipe as the result of food recognition. Furthermore, the nutrition content of a dish can be estimated from the ingredients listed in the recipe. As showed in Figure 1, recipe is textually rich of food information, with nouns referring to ingredients, numeric numbers indicating food quantities in units such as gram, and verbs pointing to cutting and cooking styles. Thus, recipes provide various aspects of cues in quantifying nutrition content of dishes even if FCT is not available.



	Recipe
Title	Shrimp Lemon Pepper Linguini
Ingredients	1 (8 ounce) package linguine pasta; 1 tablespoon olive oil; 1 pound fresh shrimp, peeled and deveined; 1 lemon, juiced; 3 tablespoons chopped fresh parsley…
Instructions	1.Bring a large pot of lightly salted water to a boil. Add linguine and cook until al dente; 2.Heat oil and saute garlic about 1 minute. Mix in chicken broth, wine, lemon juice, lemon zest, salt, and pepper. Reduce heat, and simmer until liquid is reduced by about 1/2. 3.Mix shrimp, butter, parsley, and basil into the saucepan. Cook 2 to 3 minutes, until shrimp is opaque. Stir in the cooked linguine, and continue cooking until well coated.

Fig. 1: An example of recipe-image pair from the cooking website “AllRecipe”.

Instead of demanding clean annotations, the deep models for image-to-recipe retrieval [13], [14], [15] can be trained with recipe-image pairs freely crawled from the cooking sharing websites like “AllRecipe” [1], “Cookpad” [2] and “Xiachufang” [3]. The training objective is to learn image and recipe features such that their distance is close if belonging to the same pair. Nevertheless, the recipe-image pairs are uploaded by amateur without quality control. The images associated with a recipe could be the dish under preparation or even the utensil employed for cooking the dish (see Figure 4). Furthermore, the recipes are always written in free-style. The same ingredient might be described with different names possibly mingled with terms highlighting the way that an ingredient is cooked and cut. Due to these issues in data noise, the trained model is ineffective in capturing the multi-modal correspondence between images and recipes. As a consequence, despite using tens of thousands of recipe-image pairs for model training, the state-of-the-art models only show satisfactory performance on small datasets, ranging from one thousand to ten thousands

Bin Zhu and Wing-Kwong Chan are with the Department of Computer Science, City University of Hong Kong, Hong Kong 999077, China (e-mail: binzhu4-c@my.cityu.edu.hk).

Chong-Wah Ngo is with School of Information Systems, Singapore Management University, Singapore 188065.

[1] https://www.allrecipes.com

[2] https://cookpad.com/

[3] https://www.xiachufang.com/

of recipes [14], [16], [17], [18], [19], and at most fifty thousands [20].

In reality, as most recipes are associated with sample images, food recognition can be attained by simply searching the similar sample images of a query food image. In other words, the recipe can be retrieved by comparing the query with the sample images of recipes, rather than with recipe features obtained from the model trained with noisy recipe-image pairs. In this paper, we aim to answer the following fundamental questions:

1. To what extent could the noisily trained cross-modal image-to-recipe (I2R) retrieval achieve in comparison to the more straightforward image-to-image (I2I) retrieval?

2. How far does the text-rich but noise-oriented recipes supplement the image content for retrieval?

3. How far could the current state-of-the-arts trained with webly labeled pair data scale-up to a dataset beyond ten thousands of recipes?

We derive three groups of features for experiments: image features learnt using images only; image features learnt using recipe-image pairs; recipe features learnt using recipe-image pairs. Given a food image as a query and a collection of recipe-image pairs as a dataset, we contrast the performance between I2I and I2R retrieval using three groups of features. In I2I, only the food images associated with recipes are compared with query images. The recipes are ranked based on the similarity scores of image comparison. In I2R, a query image is compared directly with recipes for similarity ranking without involving their associated images. Along the study, we also investigate the number of image samples per recipe that should be involved in model training and retrieval. The number represents the trade-off between the amount of information that a model can learn and the noises that introduced when more images are participated in training. The empirical comparison is conducted on Recipe1M [14], the largest recipe dataset available to-date, with recipes crawling from more than 20 popular recipe sharing websites.

II. Related works

Food recognition aims to recognize the category and food attributes, such as ingredient composition, cooking and cutting methods of a dish, which facilities various health related applications, ranging from food intake tracking [21], [22], nutrition estimation [23], [24] to meal recommendation [25], [26]. In the literature, the three main approaches for food recognition are: model-based recognition [7], [9], [27], context-based retrieval [28], [29] and cross-modal retrieval [13], [14], [20], [18]. We thus review the previous works from the three aspects in this section.

A. Model-based Recognition

Model-based recognition methods train a recognition model with clean human-annotated category and ingredient labels. A classic pipeline is to extract hand-crafted features such as SURF [30], [31], [7], color histogram [32], [33], [31], [7] and SIFT [34], [32], [33] then employ a classifier such as SVM [35], [31], [33] and random forests [36], [7] for recognition. Nevertheless, the performance using this pipeline is still limited and unsatisfactory. With the rapid advancement of deep learning [12], model-based recognition of food can achieve as close as 90% in accuracy on the benchmark datasets collected from different geography regions. These datasets include Western (Food-101 [7]), Japanese (UEC-256 [8]) and Chinese food (Vireo-172 [9] and Vireo-251 [37]).

The efforts of using deep learning for food recognition range from multi-task learning [9], [11], [37], architecture design [27], [38], zero or few-shot learning [39], [40] to mobile app development [22], [41]. For example, a previous work [11] formulates food recognition as a problem of multi-task problem, simultaneous prediction of ingredient composition, cutting and cooking attributes. In [27], a slice convolution block is proposed to specifically capture the vertical layer structure of food. Recently, few-shot learning are explored for food recognition in [40] by combining the category-level and ingredient-level features to compute the relation score between images. With these progresses made, a dietary tracking system is developed in [22] for food logging and lifestyle behaviour analysis. However, the major drawback of these approaches is the demand for huge amount of labeled data for model training. Scaling food recognition to several thousands of categories remains unexplored due to long-tail distribution of food categories in most regions.

B. Context-based Retrieval

Different from model-based recognition methods, context-based retrieval exempts from clean human annotations by leveraging query as example to retrieve the best-match food images for dish recognition. Context, especially GPS, is leveraged to narrow the search scope to trade for retrieval accuracy [28], [29]. Different from model-based recognition, the number of training images per dish category can be as small as 15 to guarantee satisfactory retrieval, as claimed in [29]. Context-based retrieval, however, is only applicable to restaurant food. In many countries, the nutrition information of restaurant food is not publicly available due to commercial reason. This limitation hinders the wide use of context-based retrieval in commercial food apps.

C. Cross-modal Retrieval

Cross-modal retrieval is a new branch of query-by-example, by retrieving the recipes of query images [13], [14]. As recipes reveal not only dish names but also ingredients and cooking styles, the ability to retrieve recipes for food can be referred to as a “kill multiple birds with one stone” strategy in food recognition. Different from recognition model and context-based retrieval, the required training samples are webly crawled image-recipe pairs. Due to the fact that recipe-image pairs are user generated content, both recipes and images are not guaranteed to be noise free. The cross-modal learning focuses on projecting the pairs of different modalities into a latent space for similarity measure. The first cross-modal retrieval modal was published in [13] using stacked attention network (SAN) [42]. SAN embeds ingredient tokens extracted from recipes by aligning to salient image regions learnt by attention mechanism. However, as only one projection is learnt for similarity measure, the modal requires online extraction of recipe features and can only support sequential search. A more sophisticated version, joint embedding learning (JNE), was proposed in [14] to align image features with both ingredients and cooking instructions. In JNE, bi-directional and hierarchical LSTMs are employed to embed word list (i.e., ingredients) and sentence sequence (i.e., cooking instructions). Compared to [13], JNE can differentiate between dishes using the same ingredients but different cooking styles. In addition, as two projections are learnt respectively for image and recipe embeddings, the features can be offline indexed for online retrieval.

JNE [14] was later improved by ATTEN [17], AdaMine [16], R^{2}GAN [20], ACME [18] and MCEN [19]. ATTEN extends JNE by modeling the significance of words and sentences to image features. The approach is efficient in masking out irrelevant words to visual food content, such as “home-made” and “classic”. As recipes are written in free-form by amateurs, ATTEN is also more robust to different writing styles of recipes. AdaMine [16] improves JNE by triplet ranking loss and adaptive learning. The former, particularly, enforces the distance between positive and negative recipe-image pairs with a margin, such that fine-grained cooking information can be effectively modeled in feature embedding.

The more recent works, R^{2}GAN [20] and ACME [18], are built upon adversarial learning. R^{2}GAN proposes a GAN learning module with one generator and two discriminators to align the recipe-image pairs in both embedding and visual space. Similar in spirit as R^{2}GAN, ACME adopts a discriminator in an adversarial way for embedding space learning. Additionally, translation consistency is enforced by the reconstruction of food image from recipe embedding and the prediction of ingredients from image embedding. Different from [20], [18], MCEN introduces stochastic variable models to explicitly compute the correlations between recipe and image for embedding learning. Among these approaches, ACME shows significantly better performances. On the subsets of Recipe1M dataset [14] with 1K and 10K recipes, the average median ranks of ACME for I2R retrieval are 1.0 and 6.7 respectively, far better than JNE, ATTEN, AdaMine and R^{2}GAN. In this paper, we adopt ACME [18] to derive the baselines for I2I and I2R retrieval for its superior performances.

III. Problem Definition

We define a recipe set as a recipe associated with at least one sample image. Denote the set as a triplet of $< r, s, c >$, where r refers to a recipe, $s = \{v_1, ..., v_k\}$ is the list of sample images with $k \geq 1$, and c is the semantic category of food. Given a food dataset $FD = \{< r_i, s_i, c_i >\}_{i=1}^{N}$ and a image query q, the problem is to search for the recipe r_q of q from FD. In image-to-image retrieval (I2I), the search is equivalent to measure the similarity between q and s_i in FD, as following:

$$I2I(q, s_i) = f(q, s_i), \tag{1}$$

where $f(\cdot)$ is an operator that fuses the similarities between q and every sample in s_i. The fusion can be based on max, average or median operator. For instance, the max operator is

$$f(q, s_i) = \max_{1 \leq j \leq k} \cos(E_q, E_{v_j}), \tag{2}$$

where E_q and E_{v_j} refer to image features in the embedding space, similarly for average and median operators which take the average or median of similarity scores. In cross-modal image-to-recipe retrieval (I2R), the retrieval of recipe is equivalent to:

$$I2R(q, r_i) = \cos(E_q, E_{r_i}), \tag{3}$$

where E_{r_i} refers to the text embedding of recipe in the latent space. In the literature, most of the research efforts are devoted to learning of compatible text and image features for cross-modal retrieval [14], [20], [18].

IV. Baselines

Figure 2 depicts the four basic networks to derive features as our baselines for comparative studies and investigating the three questions posted in Section I. The networks include single-modal learning to learn only image features, ranging from fine-tuning of CNN with semantic labels (Figure 2(a)), additional use of GAN for food image synthesis and discrimination between real and fake images (Figure 2(b)), to triplet network with semantic labels (Figure 2(c)). The architecture for cross-modal learning to learn both recipe and image features simultaneously is based on ACME [18] which performs embedding for both images and recipes while employing three different loss functions (Figure 2(d)). In this comparative study, the backbone neural networks being deployed are ResNet-50 [43], LSTM [44] or GAN [45].

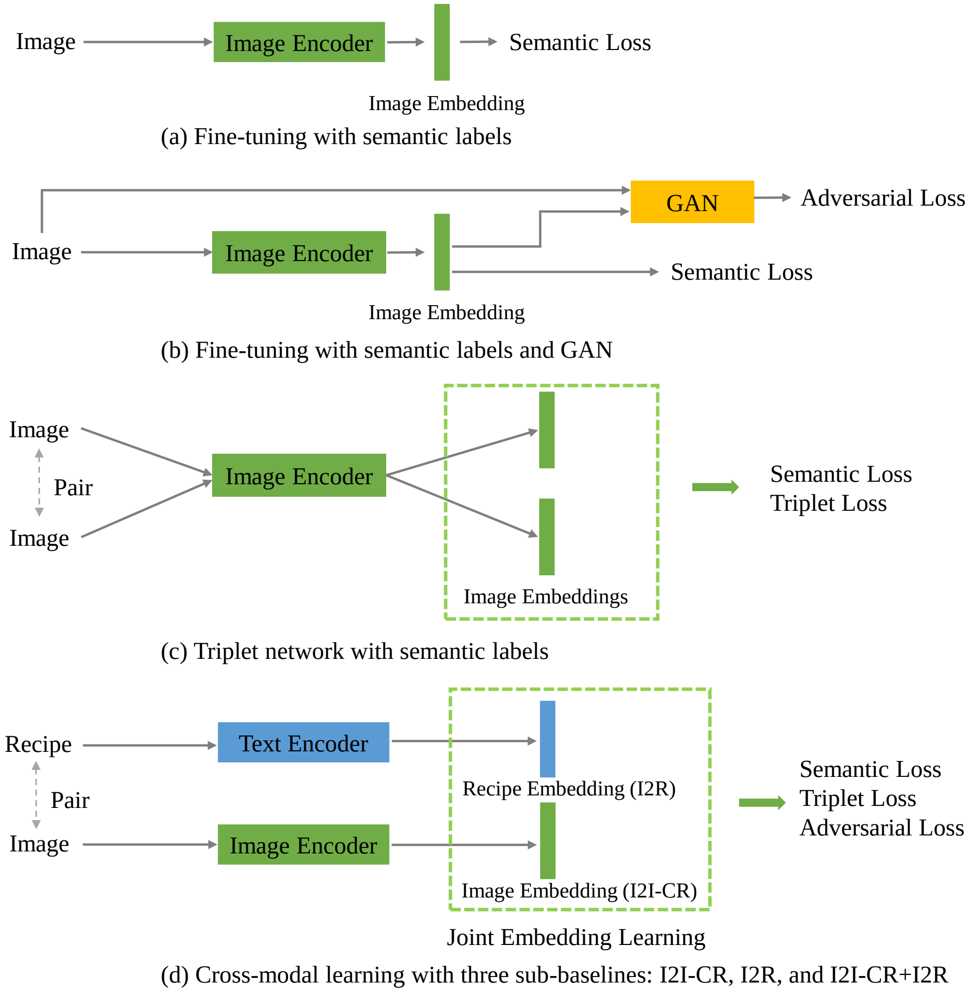
A. Single-modal Learning

Fine-tuning is a powerful way of transferring a pre-trained model to a new domain with new training examples [46]. We employ the CNN model pre-trained on Food-101 [7] with human-annotated clean labels for fine-tuning on Recipe1M dataset. In Recipe1M [14], each recipe is associated with a label indicating high-level food category. The pairs of image and label are leveraged as training samples for end-to-end model learning. Note that the category labels of Recipe1M dataset are compiled from recipe titles automatically. The resulting labels are not noise free. Furthermore, the dish titles that are not informative to be assigned to any category are simply assigned as “background” class [14]. The output layer of ResNet-50 is modified to predict the semantic labels provided to recipe sets, as shown in Figure 2(a). Entropy loss is employed:

$$L_a = -\log \frac{\exp(v_c)}{\sum_i \exp(v_{c_i})}, \tag{4}$$

where v_c is the probability of predicting ground-truth label for the input image v.

The fine-tuning model in Figure 2(a) is extended with GAN, as shown in Figure 2(b). A generator G first synthesizes an



(a) Fine-tuning with semantic labels

(b) Fine-tuning with semantic labels and GAN

(c) Triplet network with semantic labels

(d) Cross-modal learning with three sub-baselines: I2I-CR, I2R, and I2I-CR+I2R

Fig. 2: Four variants of network architectures for learning feature embeddings.

image from an image embeddingg. Similar as vanilla GAN, a discriminator D aims to distinguish between the real and fake images. The motivation of employing GAN is to train image embedding that is more discriminative for retrieval. The GAN model, as shown in Figure 2(b), is trained with both classification loss L_a and discriminator loss L_D, as following:

$$\begin{aligned} L_D =& \mathbb{E}_{x\sim p_V}[\log D(x)] + \\ & \mathbb{E}_{E_v\sim p_V}[\log(1 - D(G(E_v)))], \end{aligned} \tag{5}$$

$$L_b = L_a + \lambda L_D, \tag{6}$$

where E_v is the embedding of image v and λ is a trade-off hyperparameter to balance the two losses.

Triplet loss is reported to be effective in learning discriminative features [47]. Therefore, triplet network is also employed, as shown in Figure 2(c). Different from the first two models, this model is trained by using the recipes containing at least two sample images. In other words, the recipe sets with only one sample image cannot be involved in the training. Using Recipe1M as example, there are only 28.6% of recipe sets with more than one image. During training, the image pairs from the same recipe set are treated as positive samples, while the pairs drawn randomly from different sets are regarded as negative samples. Denote E_q, E_p and E_n as the embeddings of query, positive and negative samples respectively, the loss functions are:

$$L_{rank} = \max\{d(E_q, E_p) - d(E_q, E_n) + \alpha, 0\}, \tag{7}$$

$$L_c = L_{rank} + \beta L_a, \tag{8}$$

where $d(\cdot,\cdot)$ is a distance function to measure the similarity between image embeddings. The parameter α is margin, and β balances the relative importance of the two losses. The triplet loss (L_{rank}) aims to make the distance between positive pairs (i.e., E_q and E_p) smaller than negative pairs (i.e., E_q and E_n), while classification loss (L_a) attempts to capture high-level semantic information from the images. Following the usual practice, cosine similarity is adopted as distance function.

B. Cross-modal Learning

The architecture of Figure 2(d) is implemented based on the open source code provided by ACME [18]. Different from single-modal learning, recipe is also embedded as a feature. During training, the recipe embedding is learnt to be as

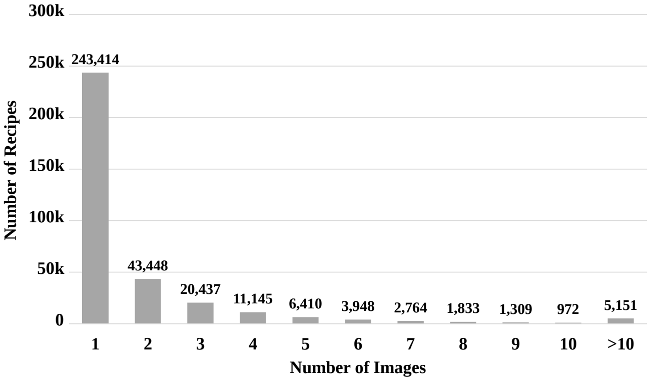


Fig. 3: Distribution of image samples size per recipe.

compatible as image embedding through three loss functions. As shown in Figure 2(d), the raw images and recipes are encoded separately and then mapped into a latent subspace for learning. Similar as fine-tuning, semantic labels of recipes are also leveraged such that category-level information are also preserved in both features during embedding. The model is trained to minimize retrieval loss $L_{retrieval}$, adversarial loss L_D and translation consistency loss L_{trans}, as following:

$$L_d = L_{retrieval} + \gamma_1 L_D + \gamma_2 L_{trans}, \tag{9}$$

where γ_1 and γ_2 control the relative importance of the losses.

The superior performance of ACME mainly attributes to retrieval loss, adversarial learning and translation consistency constraint. In Equation 9, retrieval loss $L_{retrieval}$ is based on hard-mining triplet ranking loss, which pushes positive recipe-image pairs staying close in the latent subspace, while making negative pairs far apart. Different from L_{rank} in Equation 7, rather than by random sampling, $L_{retrieval}$ chooses the most distant positive and the closest negative pairs for training. To alleviate the modality gap between recipe and image features, a discriminator is employed to predict whether an embedding stems from the recipe or image. The training is conducted in an adversarial way with the recipe encoder to align the distribution of the two modalities. The formulation of the adversarial loss is in the same spirit with Equation 5. The proposed translation consistency loss L_{trans} aims to preserve cross-modal consistency such that food image can be reconstructed from a recipe embedding while ingredient probability distribution can be predicted from an image embedding.

Based on the architecture in Figure 2(d), two baselines are derived: I2R and I2I. During retrieval, a query image is first embedded into the image latent space. I2R retrieves recipes by comparing the query embedding with the textual recipes embedded in the recipe latent space. In contrast, I2I retrieves recipes by comparing to their associated images in the image latent space.

V. Experiments

The paper aims to study three different issues in recipe retrieval for food recognition. First, performance difference between cross-modal and single-modal retrieval is compared by leveraging noisy recipe-image pairs or images alone. Second, the effect of noisy recipes in affecting retrieval performance is studied. Third, we investigate the scalability of contemporary cross-modal recipe retrieval models to data size, as well as the effect of image sample size to balance the information and noise for model training. We start by introducing experimental setting and then presenting performance evaluation along with result discussion.

Method	MedR	R@1	R@5	R@10
JNE [14]	41.9	-	-	-
ATTEN [17]	39.8	7.2	19.2	27.6
AdaMine [16]	16.5	12.5	31.5	42.2
R^{2}GAN [20]	13.9	13.5	33.5	44.9
MCNE [19]	7.2	20.3	43.3	54.4
ACME [18]	6.7	22.9	46.8	57.9
I2I (AdaMine [16])	2.0	42.9	63.8	73.8
I2I (ACME [18])	**2.0**	**43.2**	**68.7**	**78.0**

TABLE I: Performance of I2I compared with I2R of the state-of-the-art methods in 10K test set.

A. Experiment Settings

Dataset. All the experiments are conducted on Recipe1M [14], which is the only large scale public dataset with webly crawled English recipes and images. We only consider recipes with at least one image sample in our experiments. This forms a subset with 340,831 recipes and 672,580 images. Each recipe is assigned to one of the 1,048 semantic categories compiled by [14]. These labels describe the major food types of recipes, for example, “Chicken Wings”, “Creamy Blueberry Pie” and “Pizza”. Figure 3 shows the distribution for the number of image samples per recipe. The distribution is a long-tail pattern with 71.4% of recipes contain only one image. Following [18], [14], the subset is split into training, validation and testing, corresponding to 70%, 15%, 15% of recipes.

Evaluation Metrics. Median rank (MedR) and recall rate at top K (R@K) are reported to evaluate the performance. During testing, a subset is randomly sampled from the test set. Each image in the subset is regarded as a query to retrieve the corresponding recipe. MedR corresponds to the median rank position of all positions where in the rank lists the ground truth recipes of these queries are retrieved. R@K represents the percentage of true positives that are retrieved in the top K position.

Model training. The backbone network is ResNet-50 pre-trained on ImageNet. For the fine-tuning models (Figure 2(a) and 2(b)), the softmax layer of ResNet-50 is set to be equal to the number of semantic labels, which is 1,048. We follow DCGAN [48] for the implementation of GAN, except that the generator input is not noise but the global average pooling feature extracted from ResNet-50. The triplet network (Figure 2(c)) is trained using the images from recipes with more than one image. Note that the network uses 71.4% less number of training samples than other baselines. The balancing factors are set to be λ=0.1 (Equation 6), α=0.3 (Equation 7) and β=0.1 (Equation 8). The implementation of all the single-modal learning models (Figures 2(a), (b) and (c)) are based

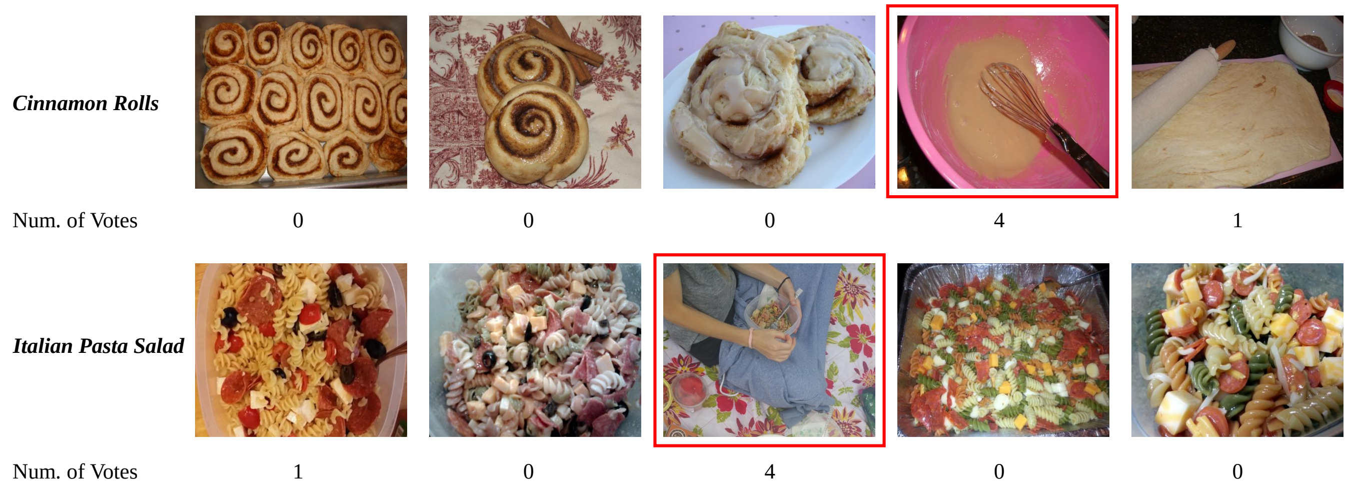


Fig. 4: Examples of noisy images excluded as queries by majority voting among the sample images in a recipe. The images receive the majority number of votes are highlighted with red bounding boxes.

	Run	Method	MedR	R@1	R@2	R@5
Single-modal	1	NFT	1,142	0.50	0.90	2.10
	2	FT	23	24.0	28.30	35.10
	3	FT+GAN	22	24.0	28.30	35.40
	4	TN	27	15.90	20.90	29.20
Cross-modal	5	I2I	7	26.70	34.23	45.72
	6	I2R	50	5.40	9.06	16.75
Ensemble	7	FT+I2I	6	32.31	38.89	49.40
	8	I2I+I2R	14	17.61	24.35	35.79
	9	FT+I2I+I2R	13	19.47	25.56	36.13
	10	I2I-All	5	32.88	39.74	50.87

TABLE II: Performance of different baselines and their combinations. "I2I-All" refers to the combinations of Runs 2, 3, 4 and 5. NFT: none fine-tuning, FT: fine-tuning, TN: triplet network, I2I: cross-modal image embedding for image-to-image retrieval, I2R: image-to-recipe retrieval.

on Adam optimizer [49], with batch size set to be 128 and initial learning rate set to be 0.0001. The model with the best MedR in the validation set are saved during training. When a model reaches plateau, the learning rate is decayed by multiplying 0.5. In GAN, the learning rates of both generator and discriminator are set to be 0.0002 initially, with a weight decay by 0.1 every 20 epochs. The implementations of cross-modal learning model (Figure 2(d)) exactly follows [18] with initial learning rate of 0.0001, and the hyperparameters γ_1 and γ_2 (Equation 9) are set to 0.005 and 0.002 respectively. The recipe encoder is composed of a LSTM and hierarchical LSTM for ingredient and instructions embedding respectively. The dimensions of both recipe and image embeddings are set to be 1024. We implement all the models with PyTorch [50] framework.

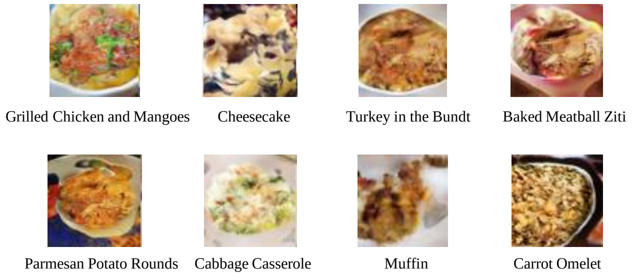


Fig. 5: Examples of images reconstructed by GAN in Figure 2b.

B. Power of I2I over I2R

To investigate the performance difference of cross-modal learning for image-to-image retrieval (I2I) over traditional image-to-recipe retrieval (I2R), we firstly present the I2I results of AdaMine [16] and ACME [18] against the contemporary state-of-the-art methods [14], [17], [16], [20], [18], [19] in Table I. Note that the testing procedure follows the conventional setting as in other works. Apparently, I2I is capable to boost the retrieval performance over I2R in a large margin in terms of both MedR and R@K. Take I2I (ACME) as an example, the MedR is improved by 70.1% from 6.7 to 2.0, similarly, the improvement of R@1 is 88.6% from 22.9 to 43.2. The result is as expected because I2I only involves single modality for retrieval, i.e., image, in contrast, I2R attempts to learn the correspondence between two modalities, i.e., text and image. The modality gap between text and image naturally makes I2R more challenging than I2I. In addition, although both recipe and image features are learnt from the unverified webly recipe-image pairs, the performance is less affected by images than the recipes written in free form format.

C. Performance Comparison

Different from [14], [17], [16], [20], [18], [51] which only involve a subset of 1K or 10K for testing, we use the whole testing set instead, involving 51,334 recipes and 82,392 images, to evaluate retrieval performances of the derived baselines. A total of 24,504 queries are selected from the recipes with more than one sample images for experiment. Note that noisy images, such as images showing the intermediate result of cooking, are excluded as queries. This is done by majority voting among the set of images in a recipe. To be specific, each image votes for the most dissimilar image in the set. Based on the Pareto's principle, the top 20% of images which receive the highest votes are not considered as queries. Figure 4 shows two representative examples of excluded noisy queries by majority voting. In the first example, i.e., the "Cinnamon Rolls", the first three sample images are final prepared dishes, while the last two are processing images showing intermediate steps of food preparation. Majority voting is able to distinguish between these two group of samples. In the second example, i.e., the recipe "Italian Pasta Salad", there is no such processing images as the first example, however, the third sample receives the majority votes since it mostly contains the background and surroundings of the dish. In the experiment, max operator (Equation 2) is used unless otherwise stated.

Table II summarizes the performance of different baselines. The results are split based on single-modal, cross-modal and their ensembles. Among the baselines, NFT (no fine-tuning) is the only model without using any training data from Recipe1M. The pre-trained model is learned based on the Food-101 [7]. Compared with fine-tuning version (FT), NFT performs much worse. The network parameters of pre-trained model are overridden during fine-tuning, showing the big domain gap between Recipe1M and Food-101, despite that both datasets contain mostly western food. Our internal result indeed shows that, if using only a small amount of training data for fine-tuning, the improvement over NFT is very limited. The result shows that transfer learning is a problem required to be addressed when there are limited number of training examples. When GAN is employed with FT, slight improvement is noted. Figure 5 shows some examples of images reconstructed by GAN (Figure 2b). In most cases, the generated images manage to sketch the outline and texture of dishes. The result of Triplet Network (Run-4) is comparable with FT despite being trained with much less number of training samples. If classification loss is not employed, i.e., β=0 (Equation 8), the MedR will degrade to 111, nevertheless.

Run-5 (I2I), which uses the image embedding features trained based on cross-modal learning, shows significantly better performance than FT with MedR improves by 16 ranks. The result indicates that the image features should have been embedded with the information extracted from text-rich but noise-oriented recipes, and the additional embedded information is indeed complementary to the original image features. Nevertheless, such improvement is not observed in Run-6 (I2R), although the recipe embedding is learned to be compatible with image embedding. The performance of I2R is indeed even worse than FT. We believe that image embedding takes advantages of the fact that one recipe may have multiple images. As model training is based on triplet rank loss, the model also learns to make different images of a recipe as similar as possible in order to align them with the recipe embedding. Having multiple images per recipe can effectively alleviate the noise issue. Having a single version of recipe, nevertheless, does not allow learning of embedding robust to potential noise. While cross-modal learning is conducted by using recipe-image pairs as training examples, the recent works in [15], [52] explore inclusion of unpaired data for model training. We adopt the self-supervised recipe loss in [52] by re-training ACME with additional 482,231 number of recipes without image samples. Nevertheless, no noticeable performance improvement is observed and both versions of ACME exhibit similar I2R performance across difference scales of datasets as shown in Figure 8. To fully leverage unpaired data, we believe that the underlying network architecture needs to be revised, as demonstrated in [15], [52], which is worth further studying but is outside the scope of this paper. In particular, we believe transformers [53], [54], [55], [56], [57], [58] can potentially unite the representation learning for recipe and images. Also note that there is a big gap between the two ranking loss based methods, Run-4 and Run-5. We attributes the performance gap to two reasons. First, the size of training examples for Run-4 is much smaller than Run-5 due to the fact that only recipes with at least two sample images can be used for learning. Second, the joint learning between recipe and image has improved the feature representation of Run-5.

Figure 6 shows two standard examples of the top 3 retrieved results by I2I and I2R in cross-modal learning. In the first example (rows 3-5), I2I manages to rank the ground truth (GT) image and thus the associate recipe in the first place. I2R ranks the GT recipe rank at 128 position due to the lack of ability to discriminate a variety of recipes in making sandwiches. It can be observed that the top 3 results of I2R are all about "Sandwiches" or "Tuna Salad", which share common ingredients as the GT recipe, such as "egg". The second and third returned results of I2I both belong to the same recipe "Deviled Eggs Salad Sandwiches", which can be seen as a more similar recipe than the I2R top results in view of the common ingredients with the GT recipe and the visual appearance of dish images. In the second example (rows 6-8), I2R is able to rank the GT recipe in the first place, which is better than I2I. Although not being a common case, we can get some clues of the advantage of I2R over I2I in certain circumstances. I2I pays more attention to the visual similarities and cues, such as color and texture of the dish images. I2R is superior to I2I when the dish images of a recipe are very different in visual appearance.

Fusing multiple runs in general boosts the retrieval performance as shown in the third part (Ensemble) of Table II. Average late fusion is adopted here to combine the rank lists of different runs. The result shows that the fusion of I2I and I2R, i.e., Run-8, degrades I2I in terms of both MedR and R@K ($K \leq 5$). Furthermore, by fusing I2I and FT, i.e., Run-

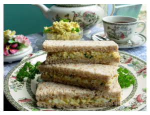
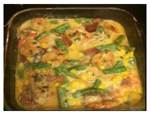
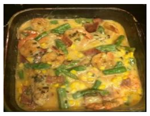
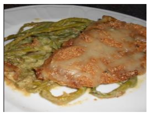
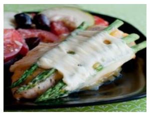
Query Image	I2I				I2R			
	Top 3 Retrieved Images and the Associated Recipes			GT Rank	Top 3 Retrieved Recipes and Paired Images			GT Rank
	**Dainty Egg and Tea Sandwiches**	Deviled Egg Salad Sandwiches		1	Egg Salad Sandwiches	Justin's Tuna Salad	Tuna Salad	128
	Egg; salad cream; mayonnaise; bread; butter; pepper…	Bread; onion; lettuce leaf; pepper; cheese; egg; mayonnaise; vinegar …			Egg; butter; bread; salt; onion; olives; celery…	Egg; mayonnaise; relish; tuna; sauce…	tuna; egg; pepper; mayonnaise; celery; lemon juice…	
	Asparagus Casserole	Chicken and Green Bean Casserole		10	**Turkey a la Oscar**	Chicken and Asparagus in Cream Soup	Cheesy Asparagus Chicken	1
	Asparagus; egg; butter; flour; milk; cheese …	Chicken breast; green beans; cheese; mayonnaise…			asparagus; water; margarine; turkey breast; shrimp …	chicken breast; asparagus; milk; bread crumb…	chicken breast; cheese; asparagus; ranch dressing…	
								

Fig. 6: Examples showing the top 3 retrieved results comparison between I2I and I2R. Given a image as query, the corresponding image or recipe ground truth (GT) is highlighted with red bounding box. The common ingredients with the GT recipe are underlined and highlighted in red.

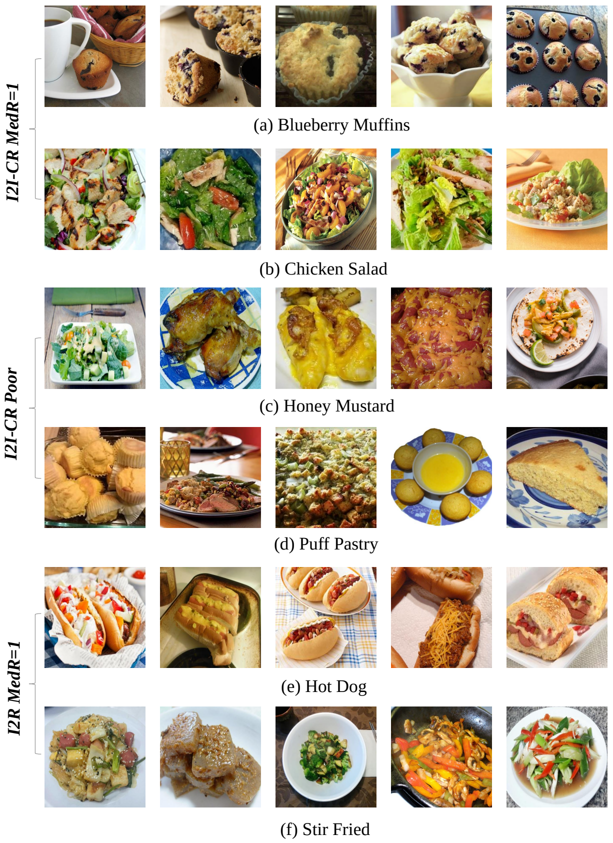


Fig. 7: Typical category samples in Recipe1M based on category-level retrieval performance.

7, better performances are observed compared with both I2I and I2I+I2R. However, further fusion with I2R, i.e., Run-9, results in worse R@K, which shows that the large modality gap between recipe and image. Although the free-form writing recipes is able to provide complement for image features in cross-modal learning, direct fusion between I2I and I2R is not beneficial. The overall best performance is obtained when fusing all I2I runs, i.e., Run-10.

We further compare the category-level retrieval performance between I2I and I2R. Among the 1,048 semantic categories, I2I outperforms I2R in 89% of the categories. In some of the categories, such as “chicken salad” (Figure 7(a)), “blueberry muffins” (Figure 7(b)) and “egg sandwich”, the MedR of I2I is as good as 1. On the other hand, I2I performs poorly when the images under a category are visually very different, for example, “honey mustard” (Figure 7(c)) and “puff pastry” (Figure 7(d)). In such cases, semantic categories cannot be properly leveraged for model training. I2R performs better than I2I for semantic categories where their food preparation processes are highly similar but different ingredients or visual appearance. Such semantic categories include “hot dog” (Figure 7(e)) and “stir fried” (Figure 7(f)), and I2R can achieve MedR=1.

D. Effect of Dataset Size

To test the scalability of the cross-modal models trained with webly crawled paired data for retrieval, we combine the training and testing examples of recipe sets as a reference set for experiment. Figure 8 shows the MedR of I2I-one (each recipe only has one image sample), I2I (each recipe can have multiple image samples) and I2R by exponentially increasing the data size from 2^{10} (1,024) to 2^{18} (262,144) of recipes.

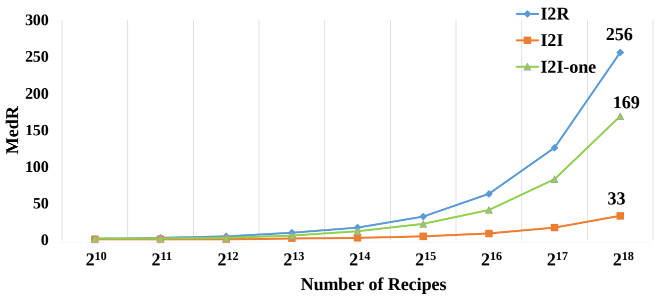


Fig. 8: Performance trend with the increase of data size.

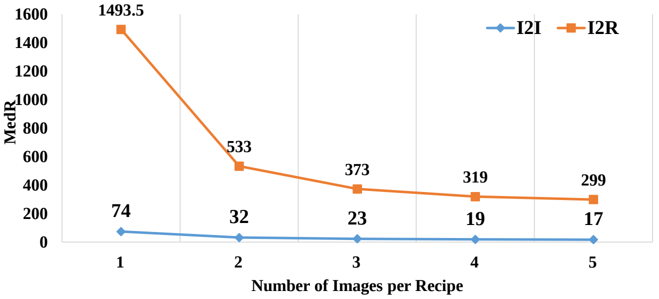


Fig. 9: The retrieval results of I2I and I2R when different number of images per recipe (x-axis) is used for model training.

Note that in I2I-one, each recipe is assumed to have only one image sample. In the experiment, for recipes having more than one image, a sample is randomly drawn for experiments. When the dataset size is small ($\leq 2^{14}$), the performances gap between I2I and I2R is negligible. However, the gap gets wider with the increase of dataset size. The learnt lessons include the followings. First, while excellent MedR results are reported on 10K recipe dataset [16], [20], [18], I2R is hardly scaled up to match the performance of I2I when dataset size is beyond 10K. We believe that the free-form writing styles of recipes is one of the critical reasons which makes the recipe features not as scalable as image features. On the other hand, for a dataset of 2^{18} recipes which is only moderately large, the MedR of I2I is already as large as 33. In other words, on average a user needs to examine the rank list till more than $30th$ rank to locate the right recipe. Third, there is a significant difference between using only one and multiple sample images for matching with query. In general, the retrieval performance is directly impacted by the number of samples in a recipe, which will be further investigated in the next section.

E. Effect of Sample Size

Sample size here refers to the number of images per recipe. We assess the impact of sample size on both training and retrieval. During training, the number of images for a recipe varies. General observation is that the more the number of images per recipe is, the better the performance will be. The experiment examines the sample size required for training an effective model. On the other hand, during retrieval, the larger the sample size for image matching is, the more likely that a recipe can be correctly retrieved. Besides, the effect of noisy sample images can be supressed with more real dish images involved in training. The experiment investigates in which extent retrieval performance is impacted by sample size.

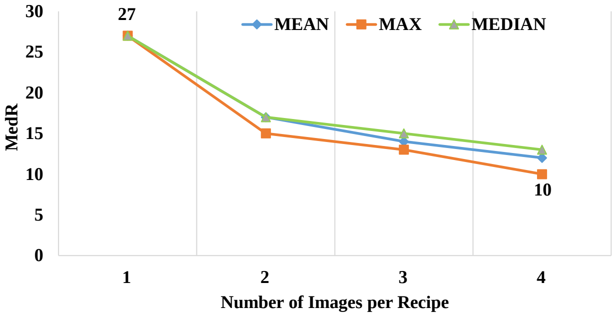


Fig. 10: The I2I result of using different operators when different number of images per recipe (x-axis) is leveraged for matching with query images.

(1) Model Training. We select 12,797 recipes each with at least five sample images for experiments. Five different cross-modal retrieval models are trained, using the sample size of 1, 2, 3, 4 and 5 respectively. In other words, each model is trained by using recipes having the same number of images. Note that the sample images are not filtered beforehand, which thus could be noisy images. The retrieval performance of the five different models are shown in Figure 9, where for each model, the MedRs of I2I and I2R are shown for comparison. Note that the experiment is conducted on the testing set of 51,334 recipes and 82,392 images (same as the Section V-C "Performance Comparison"). As shown, there is a big performance difference when using only one image and two images per recipe for training. The MedRs improve by 42 ranks for I2I and 960.5 ranks for I2R by increasing sample size from one to two images. The performance further improves gradually with the increase of sample size, but the improvement ratio in both I2I and I2R is less and less. It demonstrates that the number of images has a significant impact in calibrating the embeddings to be tolerant to noise in cross-modal training. Comparing to the results listed in Table II, despite using 94.6% less number of training examples, I2I (MedR=17) still shows better performance than methods such as FT (MedR=23) and I2R (MedR=50). In contrast to I2I, I2R needs abundant number of recipes for model training. As shown in Figure 9, I2R trained with 12,797 recipes shows fairly unsatisfactory performance (MedR=299) if compared to the model trained with 238,408 recipes (MedR=50).

(2) Recipe Retrieval. We form a reference set by pooling the recipes with at least five images for experiment. The reference set has 22,390 recipes and 89,560 images. Each image in a recipe takes turn as a query in the experiment, and the average MedR performance is reported. For each test, all the recipes in the reference set are set to have the same number of image samples. We compare the impact of sample size on MedR. The results based on I2I are shown in Figure 10. As expected, MedR improves when sample size increases, and the improvement is consistent across different operators. The degree of improvement is significant, from MedR=27 to 10 by increasing the number of sample images from one to four, when max operator is used. In other words, an effective way of boosting retrieval performance is simply by increasing the number of image samples per recipe. Among the three tested operators, retrieval based on the score from the most similar image (i.e., max operator) appears to be a good strategy. Nevertheless, we believe that mean or median operator may perform better when the sample size increases. For example, by using the mean operator, the adverse effect due to a noisy training sample can be smoothed out by averaging the similarity values. Unfortunately, limited by the long-tail distribution of image samples per recipe (refer to Figure 3), we are not able to justify this claim.

VI. Conclusion

We have investigated the potential limits of using noisy web data to train the deep models for food recognition. The result shows that, although cross-modal learning aims to learn compatible embeddings between the image and recipe features, there is a big performance gap when using these features for retrieval. Particularly, when the data size increases beyond 10K recipes, the MedR of I2R using recipe features degrades very sharply compared to that of using image features. Compared to images, the free-form writing style in recipes might have made the learning of recipe features ineffective and not scalable. Having said that, comparing between the image features learnt with recipe-image pairs and images respectively, the former shows much better retrieval performance. The result indicates that, although the paired multimedia data are noisy, the image features can still take advantage of recipe information to significantly boost retrieval performance.

To answer the three questions posted in Section I, the comparative study provides the following insights. First, the noisily trained model manages to learn effective image features but not recipe features for I2R. The recipe features perform much worse than the image features trained under the single modal setting. Second, the image features can take advantages of text-rich recipes for feature refinement, despite that the recipes are not noise-free. Hence, learning image features under cross-modal setting with recipe-image pairs is a feasible option for food recognition. Third, the performance of the current state-of-the-art models can hardly scale up to large dataset if ignoring the sample images associated with the recipes during retrieval.

Acknowledgment

The work described in this paper was fully supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 11203517).

References

[1] W. Min, S. Jiang, L. Liu, Y. Rui, and R. Jain, "A survey on food computing," *ACM Comput. Surv.*, vol. 52, no. 5, Sep. 2019.

[2] G. McKeith, *You are what you eat: The plan that will change your life*. Penguin, 2006.

[3] P. Caplan, *Food, health and identity*. Routledge, 2013.

[4] P. A. Loring and S. C. Gerlach, "Food, culture, and human health in alaska: an integrative health approach to food security," *Environmental Science & Policy*, vol. 12, no. 4, pp. 466–478, 2009.

[5] P. Fieldhouse, *Food and nutrition: customs and culture*. Springer, 2013.

[6] B. M. Newman and P. R. Newman, *Development through life: A psychosocial approach*. Cengage Learning, 2017.

[7] L. Bossard, M. Guillaumin, and L. Van Gool, "Food-101–mining discriminative components with random forests," in *European Conference on Computer Vision*. Springer, 2014, pp. 446–461.

[8] Y. Kawano and K. Yanai, "Automatic expansion of a food image dataset leveraging existing categories with domain adaptation," in *Proc. of ECCV Workshop on Transferring and Adapting Source Knowledge in Computer Vision (TASK-CV)*, 2014.

[9] J. Chen and C.-W. Ngo, "Deep-based ingredient recognition for cooking recipe retrieval," in *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 2016, pp. 32–41.

[10] M. Bolaños, A. Ferrà, and P. Radeva, "Food ingredients recognition through multi-label learning," in *International Conference on Image Analysis and Processing*. Springer, 2017, pp. 394–402.

[11] J. Chen, C.-W. Ngo, and T.-S. Chua, "Cross-modal recipe retrieval with rich food attributes," in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 1771–1779.

[12] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[13] J. Chen, L. Pang, and C.-W. Ngo, "Cross-modal recipe retrieval: How to cook this dish?" in *International Conference on Multimedia Modeling*. Springer, 2017, pp. 588–600.

[14] A. Salvador, N. Hynes, Y. Aytar, J. Marin, F. Ofli, I. Weber, and A. Torralba, "Learning cross-modal embeddings for cooking recipes and food images," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 3068–3076.

[15] B. Zhu, C.-W. Ngo, and J.-j. Chen, "Cross-domain cross-modal food transfer," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 3762–3770.

[16] M. Carvalho, R. Cadène, D. Picard, L. Soulier, N. Thome, and M. Cord, "Cross-modal retrieval in the cooking context: Learning semantic text-image embeddings," in *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2018, pp. 35–44.

[17] J. Chen, C.-W. Ngo, F. Feng, and T.-S. Chua, "Deep understanding of cooking procedure for cross-modal recipe retrieval," in *Proceedings of the 2018 ACM on Multimedia Conference*, ser. MM '18, New York, NY, USA, 2018.

[18] H. Wang, D. Sahoo, C. Liu, E.-p. Lim, and S. C. Hoi, "Learning cross-modal embeddings with adversarial networks for cooking recipes and food images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 572–11 581.

[19] H. Fu, R. Wu, C. Liu, and J. Sun, "Mcen: Bridging cross-modal gap between cooking recipes and dish images with latent variable model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 570–14 580.

[20] B. Zhu, C.-W. Ngo, J. Chen, and Y. Hao, "R2GAN: Cross-modal recipe retrieval with generative adversarial network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 477–11 486.

[21] K. Aizawa, Y. Maruyama, H. Li, and C. Morikawa, "Food balance estimation by using personal dietary tendencies in a multimedia food log," *IEEE Transactions on Multimedia*, vol. 15, no. 8, pp. 2176–2185, 2013.

[22] 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 *International Conference on Multimedia Modeling*. Springer, 2018, pp. 129–141.

[23] W. Min, B.-K. Bao, S. Mei, Y. Zhu, Y. Rui, and S. Jiang, "You are what you eat: Exploring rich recipe information for cross-region food analysis," *IEEE Transactions on Multimedia*, vol. 20, no. 4, pp. 950–964, 2017.

[24] E. Aguilar, B. Remeseiro, M. Bolaños, and P. Radeva, "Grab, pay, and eat: Semantic food detection for smart restaurants," *IEEE Transactions on Multimedia*, vol. 20, no. 12, pp. 3266–3275, 2018.

[25] S. Horiguchi, S. Amano, M. Ogawa, and K. Aizawa, "Personalized classifier for food image recognition," *IEEE Transactions on Multimedia*, vol. 20, no. 10, pp. 2836–2848, 2018.

[26] W. Min, S. Jiang, and R. Jain, "Food recommendation: Framework, existing solutions, and challenges," *IEEE Transactions on Multimedia*, vol. 22, no. 10, pp. 2659–2671, 2019.

[27] N. Martinel, G. L. Foresti, and C. Micheloni, "Wide-slice residual networks for food recognition," in *Applications of Computer Vision (WACV), 2018 IEEE Winter Conference on*. IEEE, 2018, pp. 567–576.

[28] O. Beijbom, N. Joshi, D. Morris, S. Saponas, and S. Khullar, "Menu-match: Restaurant-specific food logging from images," in *Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on*. IEEE, 2015, pp. 844–851.

[29] L. Herranz, S. Jiang, and R. Xu, "Modeling restaurant context for food recognition," *IEEE Transactions on Multimedia*, vol. 19, no. 2, pp. 430–440, 2017.

[30] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[31] Y. Kawano and K. Yanai, "Real-time mobile food recognition system," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2013, pp. 1–7.

[32] M. Chen, K. Dhingra, W. Wu, L. Yang, R. Sukthankar, and J. Yang, "Pfid: Pittsburgh fast-food image dataset," in *2009 16th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2009, pp. 289–292.

[33] S. Yang, M. Chen, D. Pomerleau, and R. Sukthankar, "Food recognition using statistics of pairwise local features," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 2249–2256.

[34] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[35] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 2013.

[36] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[37] J. Chen, B. Zhu, C.-W. Ngo, T.-S. Chua, and Y.-G. Jiang, "A study of multi-task and region-wise deep learning for food ingredient recognition," *IEEE Transactions on Image Processing*, vol. 30, pp. 1514–1526, 2020.

[38] W. Min, L. Liu, Z. Luo, and S. Jiang, "Ingredient-guided cascaded multi-attention network for food recognition," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 1331–1339.

[39] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 10 542–10 550.

[40] S. Jiang, W. Min, Y. Lyu, and L. Liu, "Few-shot food recognition via multi-view representation learning," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 16, no. 3, pp. 1–20, 2020.

[41] D. Sahoo, W. Hao, S. Ke, W. Xiongwei, H. Le, P. Achananuparp, E.-P. Lim, and S. C. Hoi, "Foodai: Food image recognition via deep learning for smart food logging," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 2260–2268.

[42] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, "Stacked attention networks for image question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 21–29.

[43] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[44] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[45] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[46] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in neural information processing systems*, 2014, pp. 3320–3328.

[47] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[48] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[49] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations*, 2015.

[50] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, vol. 32, pp. 8026–8037, 2019.

[51] B. Zhu and C.-W. Ngo, "CookGAN: Causality based text-to-image synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5519–5527.

[52] A. Salvador, E. Gundogdu, L. Bazzani, and M. Donoser, "Revamping cross-modal recipe retrieval with hierarchical transformers and self-supervised learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 475–15 484.

[53] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[54] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[55] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019, pp. 4171–4186.

[56] Y.-C. Chen, L. Li, L. Yu, A. El Kholy, F. Ahmed, Z. Gan, Y. Cheng, and J. Liu, "Uniter: Universal image-text representation learning," in *European conference on computer vision*. Springer, 2020, pp. 104–120.

[57] H. Zhang, Y. Hao, and C.-W. Ngo, "Token shift transformer for video classification," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021.

[58] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," *arXiv preprint arXiv:2103.00020*, 2021.

Bin Zhu (Member, IEEE) received the B.E. degree from Southeast University, Nanjing, China, in 2015, and the M.E. degree from Zhejiang University, Hangzhou, China, in 2018. He is currently a Ph.D. candidate with VIREO Group, department of computer science, City University of Hong Kong. His research interests mainly lie in diet tracking, generative model and multimedia analysis, including food recognition, cross-modal recipe retrieval, nutrition estimation and image generation.

Chong-Wah Ngo (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees in computer engineering from Nanyang Technological University, Singapore, and the Ph.D. in computer science from the Hong Kong University of Science and Technology (HKUST), Hong Kong. He is currently a Professor with the School of Information Systems, Singapore Management University, Singapore. Before joing Singapore Management University, he was 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, the University of Illinois at Urbana-Champaign (UIUC), Urbanna, 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. Prof. Ngo was the Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA (2011-2014). 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 Program Co-Chair of ACM Multimedia 2019, ACM Multimedia Modeling 2012 and ICMR 2012. He was the Chairman of ACM (Hong Kong Chapter) from 2008 to 2009.

Wing-Kwong Chan (Member, IEEE) received the B.Eng. degree in computer engineering from the University of Hong Kong, Hong Kong, China, and the master degree and Ph.D. degree from the University of Hong Kong, Hong Kong, China. He is an Associate Professor with the City University of Hong Kong, Hong Kong. He is the Special Issues Editor for Journal of Systems and Software. He has authored or coauthored more than 100 papers in venues such as ACM Transactions on Software Engineering and Methodology, IEEE TRANSACTIONS ON SOFTWRE ENGINEERING, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON SERVICE COMPUTING, IEEE TRANSACTIONS ON RELIABILITY, Communications of the ACM, Computer, International Conference on Software Engineering, ACM SIGSOFT Symposium on the Foundation of Software Engineering, International Symposium on Software Testing and Analysis, International Conference on Automated Software Engineering, International World Wide Web, International Conference on Web Services, and International Conference on Distributed Computing Systems. His current main research interests include program analysis and testing for concurrent software and systems.