Learning to Make Better Mistakes: Semantics-aware Visual Food Recognition

Hui Wu
IBM Research
wuhu@us.ibm.com

Michele Merler
IBM Research
mimerler@us.ibm.com

Rosario Uceda-Sosa
IBM Research
rosariou@us.ibm.com

John R Smith
IBM Research
jsmith@us.ibm.com

ABSTRACT

We propose a visual food recognition framework that integrates the inherent semantic relationships among fine-grained classes. Our method learns semantics-aware features by formulating a multi-task loss function on top of a convolutional neural network (CNN) architecture. It then refines the CNN predictions using a random walk based smoothing procedure, which further exploits the rich semantic information.

We evaluate our algorithm on a large "food-in-the-wild" benchmark [3], as well as a challenging dataset of restaurant food dishes with very few training images. The proposed method achieves higher classification accuracy than a baseline which directly fine-tunes a deep learning network on the target dataset. Furthermore, we analyze the consistency of the learned model with the inherent semantic relationships among food categories. Results show that the proposed approach provides more semantically meaningful results than the baseline method, even in cases of mispredictions.

Categories and Subject Descriptors H.3.3 [**Information Systems**]Information Storage and Retrieval-*Content Analysis and Indexing*; I.4 [**Computing Methodologies**]:Image Processing and Computer Vision
General Terms Hierarchical Deep Learning
Keywords Food Recognition, Multi-task Learning

1. INTRODUCTION

Food recognition has recently attracted a lot of attention in the multimedia and vision community, following the deluge of food pictures shared on the web and social media (either generic[1] or highly specific ones[2]). Meanwhile, with the growing popularity of fitness applications, the need for easy logging of calorie consumption on mobile devices has become increasingly relevant. While some preliminary work has been proposed to address the problem of nutrition assessment from images [10, 12], a full solution has yet to be achieved. We focus on developing a food recognition engine which can be used as a fundamental building block towards such applications for automatic nutrition intake tracking.

Sushi
CNN-FT: Chicken Wings
CNN-HL-LI: Sashimi

Cheesecake
CNN-FT: Croque Madame
CNN-HL-LI: Strawberry Shortcake

Ravioli
CNN-FT: Apple Pie
CNN-HL-LI: Dumplings

Figure 1: Examples of misclassified food items using CNN-FT and the proposed CNN-HL-LI. Ground truth labels are in green. While both methods misclassify the images, the predictions from CNN-HL-LI are semantically closer to the ground truth.

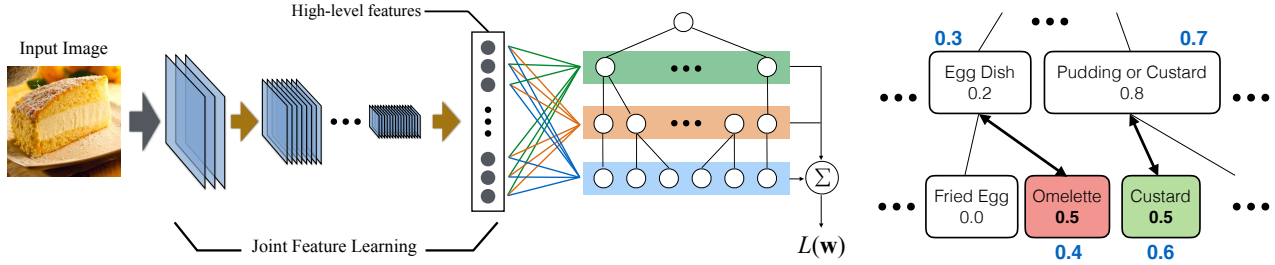
A range of different approaches have been proposed to address food recognition as an instance of a fine-grained classification problem [22], from random forests [3] to structured SVMs on top of extreme learning machines [13], from directly training or fine-tuning deep convolutional neural nets from food images [6, 9, 19] to using image captioning techniques [8]. When the geolocation information of a food picture is available, the scope of the problem is reduced from the recognition of food "in the wild" to matching items on a restaurant menu to enhance the recognition performance[1, 2, 17]. However, treating food recognition as a flat fine-grained classification problem disregards the rich semantic relationships among food classes. As shown in the examples in Figure 1, even when a food recognition engine fails to identify the exact class, it is preferable for a user if the system provides results consistent with the ground truth at a coarser category level. Based on this observation, we propose a deep CNN based food recognition framework that is aware of the semantic structured relationships among classes.

Multi-task learning has been extensively employed to op-

[1]www.instagram.com
[2]www.foodspotting.com, www.yummly.com

MM '16, October 15-19, 2016, Amsterdam, Netherlands

DOI: http://dx.doi.org/10.1145/2964284.2967205



(a) Each semantic level has one separate fully connected layer. (b) The refined labels are in blue.

Figure 2: The proposed approach consists of (a) joint feature learning, where the semantic label hierarchy is embedded in the loss function, and (b) label refinement by explicitly utilizing relationships among classes.

timize for problems with a structured output, including as in our case a hierarchical one [5, 7]. However, such an approach is relatively new within the context of deep convolutional networks [20]. In terms of encoding label structures with CNNs, the proposed approach is similar to some existing techniques. Label structures within CNNs have been modeled as dedicated fully connected layers feeding each other (from coarse to fine) [18], and graphs linking fine-grained classes to auxiliary categories encompassing label or attribute relations [4, 16, 22]. Instead of class relationships, Zhang et al. [21] learn feature representations by jointly optimizing the fine-grained classification and hierarchical similarity constraints. However, most of those proposed approaches, while "deep" in network architecture, tend to be "shallow" in semantic hierarchy (they mostly handle a two-level hierarchy). Moreover, such approaches explicitly model semantic links between each pair of coarse and fine classes (for example, using triplet loss [21] or hierarchical weight regularization [16, 22]), which requires an extremely large amount of training examples and increases optimization complexity. Our proposed architecture, on the other hand, can easily generalize to an arbitrary number of levels in the hierarchical tree without demanding a substantial increase in the number of training examples. The main contributions of this work include:

- A flexible multi-task loss function that jointly learns CNN features across different semantic levels, which can be easily integrated with different CNN designs.
- A random walk based label refinement strategy that takes advantage of the semantic structure to improve consistent predictions at each semantic level.

2. PROPOSED METHOD

As shown in Figure 2, the proposed approach consists of two components, both utilizing the hierarchical semantic relationships among classes: a semantics-aware joint feature learning step based on multi-task learning, and a random walk based label refinement step.

2.1 Semantics-aware Joint Feature Learning

Given N training images, $\{\mathbf{x}_i, y_i\}_{i=1}^N$, where $\mathbf{x}_i$ is the vectorized pixel values of an image, and $y_i \in \mathcal{Y}$ is the corresponding image label, the classical softmax loss function used in CNN is formulated as:

$$L(\mathbf{w}) = \sum_{i=1}^{N} -\log\, p(y_i|\mathbf{x}_i, \mathbf{w}) \tag{1}$$

where $\mathbf{w}$ is a vector composed of the network parameters, and $p(y_i|\mathbf{x}_i, \mathbf{w})$ is the output at the y_i-th node in the output layer. While the above formulation applies to general fine-grained classification, it does not explicitly model structures such as semantic relationships among classes.

Suppose we are provided with a semantic structure, $\mathcal{T} = \{\mathcal{Y}^{(t)}\}_{t=0}^{T}$, where $\mathcal{Y}^{(t)} = \{0, 1, \cdots, c^{(t)}\}$ is the set of $c^{(t)}$ labels at the t-th level of the semantic hierarchy. $\mathcal{Y}^{(T)}$ is the root node, which represents a generic concept of "food". The leaf nodes, $\mathcal{Y}^{(0)} \equiv \mathcal{Y}$, correspond to the original input classes. By leveraging the semantic structure, $\mathcal{T}$, our goal is to learn high-level CNN features which can improve on fine-grained classification. Specifically, we propose the following multi-task learning based loss function:

$$\begin{aligned} L(\mathbf{w}) = &\sum_{i=1}^{N} -\log\, p(y_i^{(0)}|\mathbf{x}_i, \mathbf{w}_0, \mathbf{w}^{(0)}) \\ &+ \lambda \cdot \sum_{t=1}^{T-1} \sum_{i=1}^{N} -\log\, p(y_i^{(t)}|\mathbf{x}_i, \mathbf{w}_0, \mathbf{w}^{(t)}) \end{aligned} \tag{2}$$

where $y_i^{(t)} \in \mathcal{Y}^{(t)}$ is the corresponding class label of image i at the t-th semantic level, $\mathbf{w} = [\mathbf{w}_0, \mathbf{w}^{(1)}, \cdots, \mathbf{w}^{(T)}]$, $\mathbf{w}_0$ represents the network parameters from the input layer to the high-level feature level, and $\mathbf{w}^{(t)}$ corresponds to the parameters of the fully-connected layer that maps from the high-level features to the output layer at the t-th level of the semantic tree. λ controls the trade-off between the contribution of the fine-grained classification from the leaf level, and the semantic relationships among super categories.

Given a semantic tree (Figure 3 shows the hierarchy used in this work), Equation 2 jointly optimizes the softmax output at each level of the hierarchy. As illustrated in Figure 2(a), the feature learning module is shared by different softmax output layers, each modeling a different level in the hierarchical tree. The joint feature learning design serves two purposes: (1) it ensures that high-level CNN features for fine-grained classification are also discriminant for super categories, and (2) it helps the model to produce more semantically coherent predictions. In fact, even in cases of misclassification, its predictions tend to fall within semantically related categories (see the examples in Figure 1).

During the learning process, the weights $\mathbf{w}_0$ of the shared feature layers were initialized using the values of a network pre-trained on ImageNet, while we initialize the fully connected layers using the uniform distribution. For each fully connected layer, the weight update is only related to its corresponding softmax term; for the previous layers, the gra-

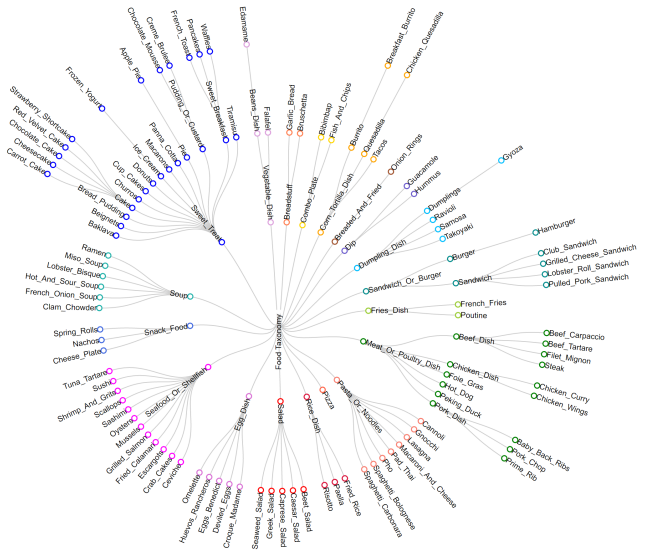


Figure 3: The food semantic hierarchy.

dient can be computed as the weighted sum of the gradient related to each loss term in Equation 2. We optimize Equation 2 using standard stochastic gradient descent.

2.2 Label Inference in the Semantic Tree

Let $\mathbf{p} = [\mathbf{h}^{(0)\intercal}, \mathbf{h}^{(1)\intercal}, \cdots, \mathbf{h}^{(T-1)\intercal}]^\intercal$ be the probability outputs at each semantic level of the learned CNN, we propose to further incorporate the semantic relationships by inferencing the probability values on the semantic tree. We formulate the label inference problem as a random walk process: the initial probability scores are considered as initial values on each node. Each node takes random transitions to its parent or child in the semantic graph at each step until the node values converge. The goal is to incorporate the recognition results from super categories as well as from the leaf nodes, so that in difficult recognition cases, information at different semantic levels are considered simultaneously.

The process is exemplified in Figure 2 (b). The initial CNN predictions (values inside each box) might still present some errors/confusion (both *Omelette* and *Custard* have a 0.5 score). The random walk process enforces consistency by smoothing the prediction values over classes on the same semantic path. It thus drives the score of *Custard* (the correct prediction, in green) to a larger value of 0.6, thanks to the large initial prediction of its parent *Pudding or Custard*.

Let $\mathbf{A}_{n\times n}$ be the adjacency matrix defined over all the tree nodes, $\mathcal{V} = \{\mathcal{Y}^{(0)}, \mathcal{Y}^{(1)}, \cdots, \mathcal{Y}^{(T)}\}$, and $|\mathcal{V}| = \sum_{t=0}^{T} c^{(t)}$ is the total number of tree nodes. $A_{i,j} = 1$ when i-th node in $\mathcal{V}$ is the parent or a child of the j-th node; otherwise $A_{i,j} = 0$. Given the adjacency matrix, we construct a transitional probability matrix $\mathbf{T}$ by normalizing each column of $\mathbf{A}$ to have unit $L1$ norm. At the k-th step of the random walk process, the current node values are determined by both the initial input values, $\mathbf{p}_0$ (normalized to have unit $L1$ norm), and the previous values on each node:

$$\mathbf{p}^{(k)} = \alpha \cdot \mathbf{p}^{(0)} + (1-\alpha) \cdot \mathbf{T}^\intercal \mathbf{p}^{(k-1)} \quad (3)$$

where α is a trade-off parameter between preserving input node values and semantic based label inference. Using the constraint of unit $L1$ norm of $\mathbf{p}^{(k)}$, Equation 3 becomes

$$\begin{aligned} \mathbf{p}^{(k)} &= \alpha \cdot \mathbf{p}^{(0)} \mathbf{1}^\intercal \mathbf{p}^{(k-1)} + (1-\alpha) \cdot \mathbf{T}^\intercal \mathbf{p}^{(k-1)} \\ &= \mathbf{T}'^\intercal \mathbf{p}^{(k-1)} \end{aligned} \quad (4)$$

where $\mathbf{T}'^\intercal = \alpha \cdot \mathbf{p}^{(0)} \mathbf{1}^\intercal + (1-\alpha) \cdot \mathbf{T}^\intercal$. The final node values are taken as the stationary value of $\mathbf{p}^{(k)}$ as $k \to \infty$. It is readily seen that the stationary node values are proportional to the eigenvector (corresponding to the largest eigenvalue) of $\mathbf{T}'$. To calculate the stationary node values, $\mathbf{p}^*$, we used the Power Method [11] by continually multiplying $\mathbf{T}'$ to the node values until convergence. In each iteration, we apply $L1$ normalization to $\mathbf{p}^{(k)}$. The final prediction for an input image is obtained by taking the label with the largest value on the sub-vector in $\mathbf{p}^*$ corresponding to the leaf nodes.

Dataset		Classes	Images per class
Food-101		101	1000
5-Chain	Applebee's	50	8
	Dennys	56	6
	Olive Garden	55	8
	Panera Bread	79	28
	TGI Fridays	54	8

Table 1: Distribution of the experimental datasets.

3. EXPERIMENTS

3.1 Datasets and Experiment Setup

To thoroughly evaluate the proposed approach, we chose two types of datasets with different properties: (1) Food-101 [3]: a benchmark food dataset commonly used to evaluate food classification accuracy [12]; (2) 5-Chain: containing menu items from five popular restaurant chains among the top causal dining in the US. The images were collected from multiple web sources, such as search engines (Bing, Google) and social networks (Flickr, Foodspotting, Yelp), and manually labeled. As shown in Table 1, categories in the 5-Chain dataset contain very few images compared to Food-101, since they are very specific. On the other hand, Food-101 contains generic food classes with various composition styles, whereas 5-Chain food items have more standard food composition and less intra-class variance (Figure 4).

For Food-101, we trained a 101-way classifier; in 5-Chain, we trained a separate classifier for each restaurant independently. We constructed a generic food taxonomy (shown in Figure 3), and for each classification task, we only take into account the subset of the taxonomy that covers the corresponding food classes. For all the experiments, we randomly selected 75% of the images for training and the rest for testing. We adopted GoogLeNet [14] as the base network for the proposed hierarchical food recognition approach. Note that our framework is independent of the base deep architecture on top of which it is applied, and could be combined with any common CNN model.

We compared against a baseline model consisting of a pre-trained GoogLeNet fine-tuned on each dataset (**CNN-FT**). This configuration has been proven to produce state of the art performance on the Food 101 dataset [12][3]. We denote

[3]Note that while we used the same network configuration as [12], the performance we present in Table 2 for Food 101 does not replicate what was reported in their paper. This might

Dataset		CNN-FT		CNN-HL		CNN-HL-LI	
		Accuracy	LCA	Accuracy	LCA	Accuracy	LCA
Food-101		69.64	0.80	72.09	**0.72**	**72.11**	**0.72**
5-Chain	Applebee's	72.55	0.93	**74.18**	**0.85**	73.91	0.86
	Dennys	57.74	1.31	59.68	1.27	**60.00**	**1.26**
	Olive Garden	78.10	0.77	79.52	0.72	**80.95**	**0.66**
	Panera Bread	91.03	0.27	90.98	0.26	**91.19**	**0.25**
	TGI Fridays	73.48	0.82	77.27	0.68	**77.78**	**0.66**

Table 2: Food recognition results from all methods across different food datasets.

(a) Hummus from Food-101

(b) TGI Fridays All American Stacked Burger

Figure 4: Example images of the datasets used.

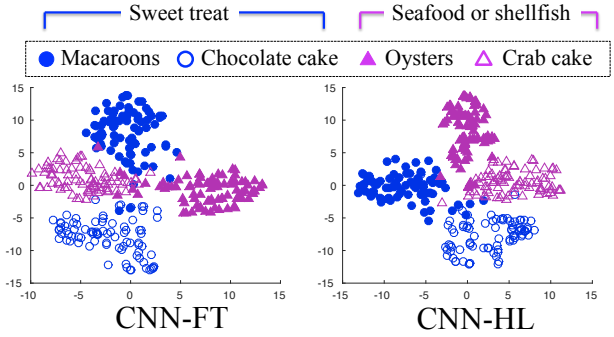


Figure 5: 2D embedding of the learned features in the proposed approach using T-SNE ([15]).

the CNN learned using the proposed multi-task *hierarchical loss* function as **CNN-HL**, and **CNN-HL-LI** with the addition of the *label inference* step.

We implemented the algorithm using the Lasagne library and the experiments were performed on a cluster with ∼20 Nvidia K40 GPUs. In all experiments, we fixed $\lambda = 1$, $\alpha = 0.2$, which were obtained using the best cross validation performance on Food-101. The random walk process is considered to reach convergence when the $L1$ norm of the node values changes by less than .001. In our experiments convergence is typically achieved within 10 iterations.

3.2 Results and Discussion

Fine-grained Food Recognition Table 2 summarizes the top-1 classification accuracy of the three methods on Food-101 and 5-Chain datasets. Except on the Panera Bread experiment, **CNN-FT** produced lower classification performance than **CNN-HL**, which demonstrates that the proposed joint feature learning strategy improves upon learning with only softmax loss at the leaf semantic layer. Furthermore, the result from **CNN-HL-LI** ranks the highest except on Applebee's. This observation indicates the effectiveness of the explicit usage of the semantics in label refinement.

Hierarchical Learning Results In addition to directly evaluating the proposed approach on fine-grained food recognition, we are also interested in the consistency of the recognition result with the provided semantic structure. We visualize the learned features by taking the output of the high-level features layer of the learned CNN and projecting them to a 2D space using T-SNE [15]. Figure 5 shows that the two super categories (*Sweet Treat*, and *Seafood or Shellfish*) are better separated with the learned CNN using the proposed multi-task loss function. This demonstrates that the joint feature learning process enables learning features that can discriminate classes at different levels of the hierarchy. Such property is especially useful for a nutrition information estimation application, since in a misclassification scenario, a semantically close prediction provides more relevant nutrition estimation than an entirely unrelated prediction.

To quantitatively evaluate whether the learned model is consistent with the semantics, we measured the semantic distance between the ground truth label and the predicted label using the height of Lowest Common Ancestor (LCA), where 0 indicates a correct match, and 2 means the two nodes share a common grandparent. We calculate the mean LCA on each dataset (shown in Table 2). The results are in line with the fine-grained classification ones, where **CNN-FT** achieves the lowest performance and **CNN-HL-LI** outperforms **CNN-HL** on the majority of the tasks. Similar observations can be made from the qualitative results shown in Figure 1. Those results demonstrate that in misclassification cases, **CNN-HL-LI** can still provide recognition results that are semantically closer to the ground truth.

4. CONCLUSIONS

We presented a framework that leverages hierarchical semantics for food recognition based on joint deep feature learning and semantic label inference. Experimental results show that the proposed framework not only improves upon the accuracy of basic flat fine-grained classification methods, but also produces more semantically coherent predictions. With the prevalence of fitness applications, such a food recognition algorithm is a first step towards visual food nutrition assessment. For future work, we plan to take advantage of the unified semantic hierarchy to transfer the knowledge learned from large datasets to small-scale ones that are challenging for fine-tuning CNNs.

be due to differences in the implementations and parameter choices, which were not released by the authors.

5. REFERENCES

[1] O. Beijbom, N. Joshi, D. Morris, S. Saponas, and S. Khullar. Menu-match: Restaurant-specific food logging from images. In *IEEE Winter Conference on Applications of Computer Vision*, pages 844–851, 2015.

[2] V. Bettadapura, E. Thomaz, A. Parnami, G. Abowd, and I. Essa. Leveraging context to support automated food recognition in restaurants. In *IEEE Winter Conference on Applications of Computer Vision*, January 2015.

[3] L. Bossard, M. Guillaumin, and L. Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision*, 2014.

[4] J. Deng, N. Ding, Y. Jia, A. Frome, K. Murphy, S. Bengio, Y. Li, H. Neven, and H. Adam. Large-scale object classification using label relation graphs. In *European Conference on Computer Vision*, pages 48–64, 2014.

[5] T. Deselaers and V. Ferrari. Visual and semantic similarity in imagenet. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1777–1784, 2011.

[6] Z. Ge, C. McCool, C. Sanderson, and P. I. Corke. Modelling local deep convolutional neural network features to improve fine-grained image classification. In *IEEE International Conference on Image Processing*, 2015.

[7] N. Goernitz, C. Widmer, G. Zeller, A. Kahles, G. Rätsch, and S. Sonnenburg. Hierarchical multitask structured output learning for large-scale sequence segmentation. In *NIPS*, pages 2690–2698, 2011.

[8] J. Hessel, N. Savva, and M. J. Wilber. Image representations and new domains in neural image captioning. *Conference on Empirical Methods in Natural Language Processing Vision + Learning workshop*, 2015.

[9] H. Kagaya, K. Aizawa, and M. Ogawa. Food detection and recognition using convolutional neural network. In *ACM Multimedia*, pages 1085–1088, 2014.

[10] Y. Kawano and K. Yanai. Foodcam: A real-time mobile food recognition system employing fisher vector. In *International Conference on MultiMedia Modeling*, pages 369–373, 2014.

[11] A. N. Langville and C. D. Meyer. A survey of eigenvector methods for web information retrieval. *SIAM review*, 47(1):135–161, 2005.

[12] A. Myers, N. Johnston, V. Rathod, A. Korattikara, A. Gorban, N. Silberman, S. Guadarrama, G. Papandreou, J. Huang, and K. Murphy. Im2calories: towards an automated mobile vision food diary. In *International Conference on Computer Vision*, 2015.

[13] C. M. Niki Martinel, Claudio Piciarelli and G. L. Foresti. A structured committee for food recognition. In *International Conference on Computer Vision*, 2015.

[14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.

[15] L. Van Der Maaten. Accelerating t-sne using tree-based algorithms. *The Journal of Machine Learning Research*, 15(1):3221–3245, 2014.

[16] S. Xie, T. Yang, X. Wang, and Y. Lin. Hyper-class augmented and regularized deep learning for fine-grained image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2645–2654, 2015.

[17] R. Xu, L. Herranz, S. Jiang, S. Wang, X. Song, and R. Jain. Geolocalized modeling for dish recognition. *IEEE Transactions on Multimedia*, 17(8):1187–1199, Aug 2015.

[18] Z. Yan, H. Zhang, R. Piramuthu, V. Jagadeesh, D. DeCoste, W. Di, and Y. Yu. Hd-cnn: Hierarchical deep convolutional neural network for large scale visual recognition. In *International Conference on Computer Vision*, 2015.

[19] K. Yanai and Y. Kawano. Food image recognition using deep convolutional network with pre-training and fine-tuning. In *IEEE International Conference on Multimedia and Expo*, pages 1–6, 2015.

[20] T. Zeng and S. Ji. Deep convolutional neural networks for multi-instance multi-task learning. In *IEEE International Conference on Data Mining*, pages 579–588, 2015.

[21] X. Zhang, F. Zhou, Y. Lin, and S. Zhang. Embedding label structures for fine-grained feature representation. *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[22] F. Zhou and Y. Lin. Fine-grained image classification by exploring bipartite-graph labels. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.