Food Recognition by Integrating Local and Flat Classifiers

Eduardo Aguilar[1,2(✉)] and Petia Radeva[2,3]

[1] Universidad Católica del Norte, Antofagasta, Chile
eaguilar02@ucn.cl
[2] Universitat de Barcelona, Barcelona, Spain
[3] Computer Vision Center, Bellaterra, Spain

Abstract. The recognition of food image is an interesting research topic, in which its applicability in the creation of nutritional diaries stands out with the aim of improving the quality of life of people with a chronic disease (e.g. diabetes, heart disease) or prone to acquire it (e.g. people with overweight or obese). For a food recognition system to be useful in real applications, it is necessary to recognize a huge number of different foods. We argue that for very large scale classification, a traditional flat classifier is not enough to acquire an acceptable result. To address this, we propose a method that performs prediction with local classifiers, based on a class hierarchy, or with flat classifier. We decide which approach to use, depending on the analysis of both the Epistemic Uncertainty obtained for the image in the children classifiers and the prediction of the parent classifier. When our criterion is met, the final prediction is obtained with the respective local classifier; otherwise, with the flat classifier. From the results, we can see that the proposed method improves the classification performance compared to the use of a single flat classifier.

Keywords: CNNs · Deep learning · Epistemic Uncertainty · Image classification · Food recognition

1 Introduction

Analysis of food images has been an emerging research topic in recent years within the field of Computer vision. Currently, there is a large number of food image datasets that makes possible to perform food recognition [2,3,5]. However, the food classes provided by these are still low compared to those needed for a real food recognition application. Just considering the most common foods, there will easily be thousands of different food classes in worldwide. On the other hand, models based on Convolutional Neural Networks (CNNs) have allowed to address problems of object recognition on a large-scale achieving promising results. Although it has also been shown that the number of classes that will be recognized and the number of images that belong to each one are inversely

A. Morales et al. (Eds.): IbPRIA 2019, LNCS 11867, pp. 65–74, 2019.
https://doi.org/10.1007/978-3-030-31332-6_6

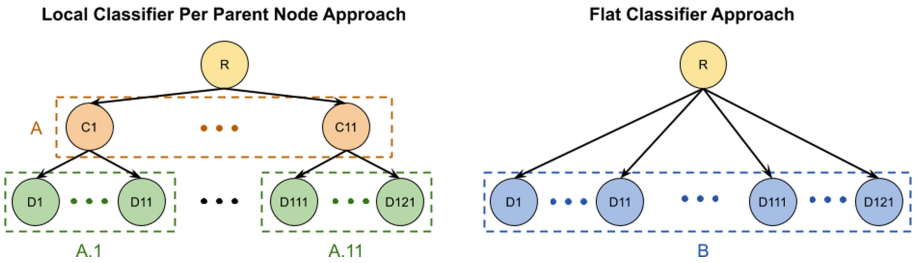


Fig. 1. Example image that illustrates the class hierarchy of the Local Classifier per Parent Node and Flat Classification approaches on MaFood-121 Dataset. A, A1-A11, B denotes the classifiers for the nodes within the respective rectangles.

proportional to the model performance [9]. Samples of this can be seen in [14], where the proposed model decreases the performance on about 10% when we compared the result on cifar-10 [7] with respect to cifar-100 [7]. In the case of food domain, a reduction of 7% is shown in [10] when we compared the result on uecfood-100 [11] with respect to uecfood-256 [5]. This suggests that the use of a single CNNs model to recognize all classes will not be enough to classify a huge amount of foods classes.

Regarding the strategies for solving classification problems, these can be grouped in two ways [12]: (1) By means of a flat classification approach, where a single classifier is used for all classes to be predicted; and (2) By means of a hierarchical classification approach, where the classes to be predicted are organized into a class hierarchy. As for the second strategy, there are three types of local classifiers to perform the predictions [12]: (2.1) Local Classifier Per Node Approach (LCN), which consists in training a binary classifier for each node of the class hierarchy; (2.2) Local Classifier Per Parent Node Approach (LCPN), which consists in training multi-class classifier for each parent node in the class hierarchy to distinguish between its child nodes; and (2.3) Local Classifier Per Level Approach (LCL), which consists of training one multi-class classifier for each level of the class hierarchy. Note that the only one hierarchical approach applied, in the context of food recognition, was proposed by [13], which incorporates in their method an LCL strategy to obtain error closer to the real class when the classification is erroneous. Focusing on the LCPN approach, the main problem is that the error in the parent local classifiers is propagated to the children. For example, in the Fig. 1, if the classifier A miss-classifies the cuisine of the image like C11, the dish recognition from the labels D111-D121 automatically will be wrong. To reduce the error propagated for the LCPN approach, we propose only to classify with this strategy those predictions that are likely well, and the remainder one to classify with a flat classification approach. To identify a good prediction, we complemented the decision of the local classifier for the parent node with the most probable child node obtained from the analysis of the Epistemic Uncertainty (EU). By definition, the EU captures the ignorance about which model generated the collected data [6]. Therefore, we expect that

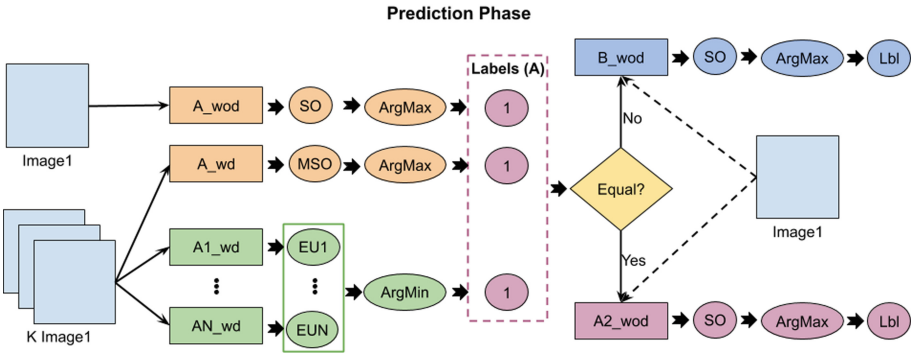


Fig. 2. Main scheme of the proposed method, which shows the procedure performed when the predictions in the first level of the hierarchy is equal to 1. The suffix wod denotes the prediction with the dropout turned off, wd denotes the prediction with the dropout turned on, the terms SO, MSO, and EU, denotes Softmax Output, Mean Softmax Output and Epistemic Uncertainty.

the correct local classifier for the child node to give a low uncertainty when the image to be predicted belongs to the classes for which it was trained.

Our main contributions in this paper are as follows: (1) we provide an epistemic uncertainty-based method, which minimizes error propagated from parents to children in the LCPN approach; (2) we propose a criterion to decide when to apply a local or flat classifier; and (3) we demonstrate that it is possible to achieve better classification results when we integrate the prediction of LCPN with flat classifier through our proposal.

The remainder of this paper is organized as follows: first, in Sect. 2, we present the proposed method; second, in the Sect. 3, we present the dataset, the experimental setup and discuss the results; finally, in Sect. 4, we describe the conclusions.

2 Proposed Method

In this section, we explain detailed the steps involved in the proposed approach, which considers local and flat classifiers to perform image predictions. In the follow subsection, we first comment the consideration in the model architecture, second we describe the equation to obtain the EU and then we explain all the components involved in our approach.

2.1 Model Setup

Every classifier trained in our proposed approach is based on the same CNNs architecture. At the top of the network, the output of the last convolution layer is flattened. Then, it is necessary to add a dropout layer after each hidden fully

connected layer so that we are able to apply the Monte Carlo dropout (MC-dropout) sampling [6] to calculate the epistemic uncertainty. Finally, it is ended up with an output layer with softmax activation and neurons equal to the number of classes. Note that all the classifiers are trained independently, and then, the proposed method is applied to give the final prediction (see Fig. 2).

2.2 Epistemic Uncertainty

During the prediction phase, a key part of our method contemplates the analysis of the EU obtained for the images during the prediction phase when we classify their with the local classifier on the second level of the hierarchy. The EU can be obtained by applying MC-dropout sampling. In practical terms, it means to predict K times the same image using the model with the dropout layer turned on. Then, the EU will correspond to the predictive entropy calculated from the K predictions given.

Formally, the EU can be expressed as follows:

Let $\overline{p(y_c = \hat{y_c}|x)}$ be the average probability that the prediction y_c is equal to the ground-truth $\hat{y_c}$ given image x, calculated from K MC-dropouts simulations. Then,

$$EU(x_t) = -\sum_{c=1}^{C} \overline{p(y_c = \hat{y_c}|x_t)} \ln(\overline{p(y_c = \hat{y_c}|x_t)}), \tag{1}$$

where

$$\overline{p(y_c = \hat{y_c}|x)} = \frac{1}{K} \sum_{k=1}^{K} p(y_c^k = \hat{y_c^k}|x). \tag{2}$$

2.3 Prediction by Integrating Local and Flat Classifiers

The proposed method contemplates the training of LCPN and also a flat classifier, and the integration of both approaches during the prediction. The proposal is thought for two level of hierarchy, but can be easily extensible to more levels.

One problem with LCPN approach is that the error is propagated from upper to lower levels. To deal with this, in our proposal, instead of directly applying the prediction given for the local parent classifier (LPC), we propose a strategy to ensure that the prediction is very likely the correct one, and thus, minimize the error propagation. This strategy consists in three parts: (a) Get the prediction of the images using the LPC with the dropout turned off; (b) Get the mean of the predictions to send K times the images to the LPC with the dropout turned on; (c) Estimate the EU of the samples for each local child classifier (LCH) and choose the label which represents the LCH that provides the lower EU for the respective image. After that, we compared the three predictions and in the case of all of them get the same value, we apply the respective LCH using the dropout turned off. Otherwise, we classify the image with the Flat classifier with dropout turned off. In Fig. 2, we illustrated the steps involves in our proposal. In this case, all the strategies (a-c) give the same prediction, and therefore, the local classifier is chosen to give the final response.

3 Experiments

In this section, we first present the dataset used, second we describe the evaluation measures, third we present the experimental setup and last we describe the results obtained with our proposed approach.

3.1 Dataset

In order to validate the benefits of our proposed method, we chose the newly published food dataset MAFood-121 [1]. This is a multi-task food image dataset comprising 3 related tasks: (a) dish, (b) cuisine, and (c) categories/food groups. The dataset was built up taking into account the top 11 most popular cuisines collecting the images from 4 different sources, 3 public food datasets [2,3,5], and Google Search Engine. For each cuisine, 11 dishes are considered with an average of 119 images per dish with their respective annotations of food categories. In total, 21.175 images were gathered, distributed as 72.5% for training, 12.5% for validation and 15% for test. For our purpose, we only consider the single label tasks (cuisine and dish) and keep the same distribution of data for training, validation and test. An example image for each cuisine can be seen in Fig. 3.

Fig. 3. An example representative image for each cuisine belonging to MaFood-121.

3.2 Metric

In order to evaluate the performance of our approach, we used the overall Accuracy (Acc), which is a standard metric for object recognition. Formally it is defined as follows:

$$Acc = \frac{1}{T}\sum_{c=1}^{C} TP_c, \tag{3}$$

where C is the number of classes, TP_c (True Positives) is the amount of images belonging to the class c classified correctly, and T is the total number of images evaluated.

3.3 Experimental Setup

We trained a CNN architecture based on ResNet-50 [4], using the categorical cross-entropy loss optimized with Adam. We modified this neural network by removing the output layer, and instead, we added one hidden fully connected

layer of 2048 neurons, followed by a dropout layer with a probability of 0.5, and we ended up with an output layer with softmax activation and neurons equal to the number of classes of the respective subset. In this particular case, there were 121 neurons for the flat classification and 11 for the local classifiers. In total, thirteen models were trained based on the same architecture. The models are named as follows:

1. *A*: CNN Model trained to perform the cuisine classification.
2. *A1–A11*: CNN Model trained to perform the local dish classification for the following cuisines: American (A1), Japanese (A2), Italian (A3), Greek (A4), Turkish (A5), Chinese (A6), Mexican (A7), Indian (A8), Thai (A9), Vietnamese (A10) and French (A11).
3. *B*: CNN Model trained to perform the flat classification for all dishes.

Model B was re-trained, from a pre-trained model on ILSVRC dataset [8], during 50 epochs with a batch size of 32, and an initial learning rate of $1e-4$. With respect to models A and A1–A11, they were re-trained on the top of the networks (after the last convolutional layer) from the pre-trained model B, during 32 epoch with a batch size of 32, and an initial learning rate of $1e-5$. In all models, we applied a decay of 0.5 every 8 epochs. On the other hand, regarding the data augmentation process, for all models, the original image is re-sized to (256, 256) and then random crops with a size of (224, 224) and horizontal flip with a 50% probability are applied. The training was done using Keras with Tensorflow as backend.

3.4 Results

In this section, we present the results obtained on the MAFood-121 dataset by the Local and Flat classification approaches and our proposed method, which integrates both during the image prediction.

One of the key elements of our approach involves calculating the EU of the sample when it is sent to several local classifiers, one for each cuisine, in order to determine the cuisine to which the image belongs to. The idea behind this is that the EU is explained with enough data. Therefore, if we have an image with the features close to those learned by a model, the EU will be small, which implies that it is very likely that this image belongs to any of their classes. Figure 4 shows an example of the results obtained in terms of EU for the test set images by three local classifiers. Each color represents a different cuisine and each row corresponds to the EU obtained for the local classifiers. Note that the EU for images belonging to the type of cuisine used for the training of the first classifier (first row) is represented with the points of the first color, for the second classifier (second row) with the points of the second color and so on. As expected, when we compared the result given for a classifier with respect to the cuisine of the image (left to right) or when we compared the results for all the classifiers with respect to the images of a specific cuisine, we can see that the EU tends to be small when the test images are similar to the images used in the training of the

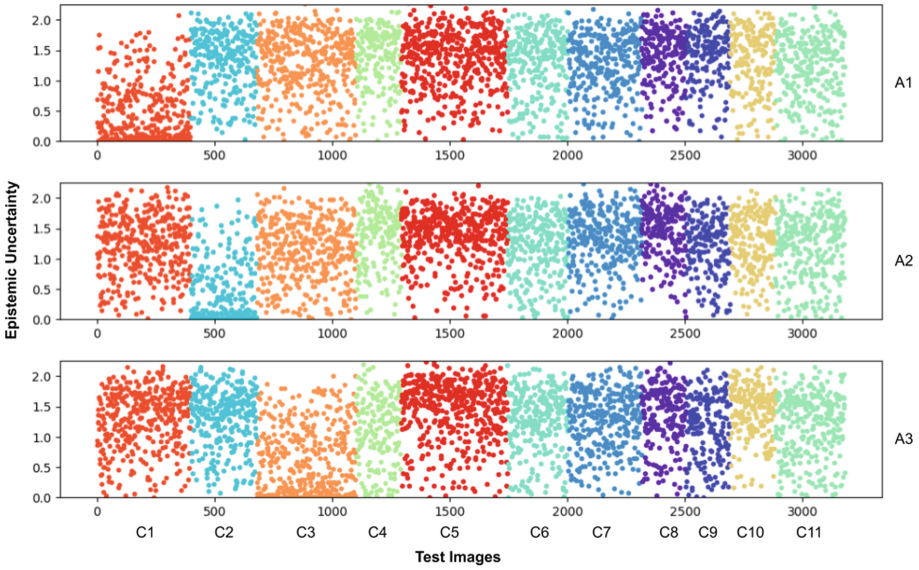
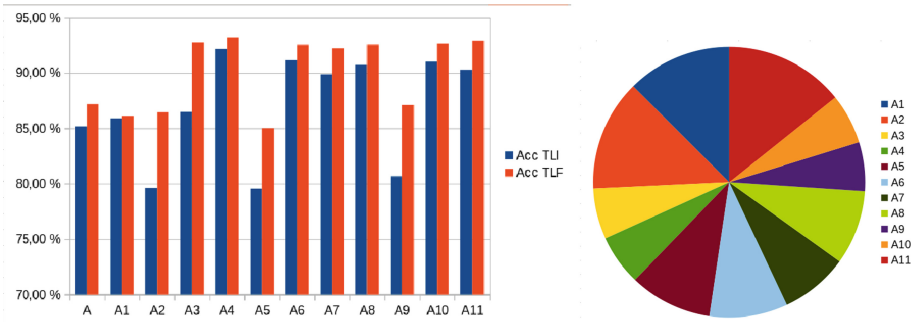


Fig. 4. Epistemic uncertainty obtained for three local cuisine classifiers in the images of the test set. Each color represents a cuisine and each subplot - the result obtained for the respective classifier. (Color figure online)

classifier. However, in some cases the minimum EU is not corresponding to the real cuisine of the image. We believe that this occurs due to the shared features among different cuisine in some cases. For this reason, we consider the analysis of the EU like a complement to the cuisine recognition classifier, instead to use directly this procedure to determine the cuisine of the image.

As for the local classifiers, we evaluate the performance applying two different training strategies: Transfer Learning from ImageNet (TLI), which consists of re-training the whole network with the food images, using as initial weight of the lower layers (before the first fully connected layer) the values obtained when the base model was trained on ImageNet dataset; and Transfer Learning from Food (TLF), which consists of freezing the lower layers and re-training only the upper layers, using as initial weight the values obtained for a model trained with the same type of data, specifically we use as a base model the flat classifier trained with all foods. The results obtained can be seen in Fig. 5a and the distribution of the test images used for each local cuisine classifiers in Fig. 5b. For all cases, using the TLF strategy we were able to improve the results of the local classifications. In particular, for the classifiers A2, A3, A6 and A9, we can see biggest increment in the performance. We believe that the improvement occurs mainly because the use of a subset of images for each classifier is not enough to avoid the overfitting of the network when we train all the layers. However, if we share the global features extracted (lower layer) from all the foods and then we only retrained



(a) (b)

Fig. 5. From left to right: (a) the accuracy obtained for each local classifier when used Transfer Learning From ImageNet (TLI) or Transfer Learning From Food (TLF) for the training of the classifier; and (b) the distribution of the test images for each cuisine.

Table 1. Results on MaFood-121 in terms of Accuracy.

Approach	Cuisine		Local		Flat		Overall
	#Images	*Acc*	#Images	*Acc*	#Images	*Acc*	*Acc*
GT+Local	3177	100.00%	3177	89.61%	-	-	89.61%
Cuisine+Local	3177	87.19%	2770	91.71%	-	-	79.96%
Flat	–	–	–	–	3177	81.37%	81.37%
Proposed method	1579	96.96%	1531	96.08%	1598	70.21%	81.62%

the last fully connected layer, the network achieves better adjustment of the model's weights.

Finally in Table 1 we show the results achieved for four approaches: (a) GT+Local, to reflect the performance of the dish classification when we have a perfect cuisine recognition; (b) Cuisine+Local, which is the base line for dish classification when we chose the local classifier per cuisine considering the cuisine recognition performance; (c) Flat, which corresponds to classification of all the classes in the same classifier; (d) Our proposed model, which integrates the Local and Flat classifiers taking into account the EU to take the cuisine decision on the image prediction. From the results, we demonstrate that it is possible to achieve large increase in terms of accuracy when we have perfect cuisine recognition and we use an individual classifier for each cuisine type (see GT+Local). In our case, the performance of cuisine recognition is far from perfect (87.19%), and for this reason, the error propagated by these predictions produced the lowest classification accuracy (79.96%) despite the local classifier per cuisine provided 91.71% of accuracy. As for our proposed, we intend to reduce the miss-classification produced by error in the cuisine recognition complementing the prediction with the results obtained by the analysis of the EU, with which 1579 images are obtained with a high likely that the cuisine is well classified. In this subset of images, we

obtain a 96.96% of accuracy on cuisine classification and 96.08% accuracy in the local classifiers per cuisine, which when combined with the Flat classifier predictions we obtain 81.62%. This result outperforms the classification obtained for the Cuisine+Local and Flat approaches.

4 Conclusions

In this paper, we proposed a new method to perform food recognition by the integration of a hierarchical with flat classifiers. In our method, we contemplated that hierarchical classification can propagate the error from parent to child nodes, and for this reason we proposed to use local classifiers only when we are sure that it is very likely that the prediction will be good. Otherwise the classification is performed with the flat classifier. To recognize good prediction, we complemented the output of the local classifier with the analysis of the EU of the images. From the results obtained, we observed that the proposed approach provides a good performance allowing us to further reduce propagation error. As a conclusion, we have shown the benefits of the proposed approach, which can be a good alternative when we have to predict a huge number of classes. As future work, we will explore the application of EU to another problems such as, novelty detection or active labeling.

Acknowledgement. This work was partially funded by TIN2015-66951-C2-1-R, 2017 SGR 1742, Nestore, Validithi, 20141510 (La MaratoTV3) and CERCA Programme/Generalitat de Catalunya. E. Aguilar acknowledges the support of CONICYT Becas Chile and M. P. Radeva is partially supported by ICREA Academia 2014. We acknowledge the support of NVIDIA Corporation with the donation of Titan Xp GPUs.

References

1. Aguilar, E., Bolaños, M., Radeva, P.: Regularized uncertainty-based multi-task learning model for food analysis. J. Vis. Commun. Image Represent. **60**, 360–370 (2019)
2. Bossard, L., Guillaumin, M., Van Gool, L.: Food-101 – mining discriminative components with random forests. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8694, pp. 446–461. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10599-4_29
3. Güngör, C., Baltacı, F., Erdem, A., Erdem, E.: Turkish cuisine: a benchmark dataset with Turkish meals for food recognition. In: 2017 25th Signal Processing and Communications Applications Conference (SIU), pp. 1–4, May 2017. https://doi.org/10.1109/SIU.2017.7960494
4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
5. Kawano, Y., Yanai, K.: Automatic expansion of a food image dataset leveraging existing categories with domain adaptation. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8927, pp. 3–17. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16199-0_1

6. Kendall, A., Gal, Y.: What uncertainties do we need in Bayesian deep learning for computer vision? In: Advances in Neural Information Processing Systems, pp. 5574–5584 (2017)
7. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Technical report. Citeseer (2009)
8. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
9. Luo, C., Li, X., Yin, J., He, J., Gao, D., Zhou, J.: How does the data set and the number of categories affect CNN-based image classification performance? J. Softw. **14**(4), 168–181 (2019)
10. Martinel, N., Foresti, G.L., Micheloni, C.: Wide-slice residual networks for food recognition. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 567–576. IEEE (2018)
11. Matsuda, Y., Hoashi, H., Yanai, K.: Recognition of multiple-food images by detecting candidate regions. In: 2012 IEEE International Conference on Multimedia and Expo, pp. 25–30. IEEE (2012)
12. Silla, C.N., Freitas, A.A.: A survey of hierarchical classification across different application domains. Data Min. Knowl. Disc. **22**(1–2), 31–72 (2011)
13. Wu, H., Merler, M., Uceda-Sosa, R., Smith, J.R.: Learning to make better mistakes: semantics-aware visual food recognition. In: Proceedings of the 24th ACM International Conference on Multimedia, pp. 172–176. ACM (2016)
14. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: Proceedings of the British Machine Vision Conference (BMVC), pp. 87.1-87.12 (2016)