 CMC, vol.59, no.1, pp.297-313, 2019

Image Augmentation-Based Food Recognition with Convolutional Neural Networks

Lili Pan[1], Jiaohua Qin[1,*], Hao Chen[2], Xuyu Xiang[1], Cong Li[1] and Ran Chen[1]

Abstract: Image retrieval for food ingredients is important work, tremendously tiring, uninteresting, and expensive. Computer vision systems have extraordinary advancements in image retrieval with CNNs skills. But it is not feasible for small-size food datasets using convolutional neural networks directly. In this study, a novel image retrieval approach is presented for small and medium-scale food datasets, which both augments images utilizing image transformation techniques to enlarge the size of datasets, and promotes the average accuracy of food recognition with state-of-the-art deep learning technologies. First, typical image transformation techniques are used to augment food images. Then transfer learning technology based on deep learning is applied to extract image features. Finally, a food recognition algorithm is leveraged on extracted deep-feature vectors. The presented image-retrieval architecture is analyzed based on a small-scale food dataset which is composed of forty-one categories of food ingredients and one hundred pictures for each category. Extensive experimental results demonstrate the advantages of image-augmentation architecture for small and medium datasets using deep learning. The novel approach combines image augmentation, ResNet feature vectors, and SMO classification, and shows its superiority for food detection of small/medium-scale datasets with comprehensive experiments.

Keywords: Image augmentation, small-scale dataset, deep feature, deep learning, convolutional neural network.

1 Introduction

In human life, food ingredients have always been essential they frequently draw the masses' much more interesting than before. At present, food-ingredient suppliers detected abundant categories of food ingredients and labeled them properly with the human visual system. This process is very tiring, uninteresting, and expensive [Chen, Xu, Xiao et al. (2017)]. Therefore, it becomes urgent to construct a food-ingredient recognition system, which can intelligently recognize food-ingredient images and label correct food categories.

Recently, image recognition implements great growth in many fields [Li, Qin, Xiang et al. (2018); Pouyanfar and Chen (2016); Chen, Zhu, Lin et al. (2013); Liu, Wang, Liu et al.

[1] College of Computer Science and Information Technology, Central South University of Forestry and Technology, Changsha, 410004, China.

[2] College of Computer Science and Electronic Engineering, Hunan University, Changsha, 410082, China.

[*] Corresponding Author: Jiaohua Qin. Email: qinjiaohua@163.com.

(2017)], such as remote sensing, digital telecommunications, medical imaging, and so on. A variety of work have shown that deep learning and machine learning technologies can be exploited to retrieve food images intelligently [Chen, Xu, Xiao et al. (2017); Pan, Pouyanfar, Chen et al. (2017); Yanai and Kawano (2015); Joutou and Yanai (2009)]. While most food recognition methods concentrate on diet [Joutou and Yanai (2009); Hoashi, Joutou and Yanai (2010); Kagaya, Aizawa and Ogawa (2014)], and food datasets are mainly made up of food meal images. Fig. 1 shows six kinds of food meals. Nowadays, few food-ingredient datasets (as shown in Fig. 2) are obtainable, and thus, the multi-category detection of food ingredients is limited in existing literature [Chen, Xu, Xiao et al. (2017); Pan, Pouyanfar, Chen et al. (2017)] and the size of obtained food-ingredient images is commonly small scale or medium. To effectively classify small and medium food-ingredient datasets, this study presents an image augmentation-based food recognition architecture utilizing deep learning.

Figure 1: Food meals

Figure 2: Food ingredients

The study [Hinton and Salakhutdinov (2006)] showed that high-dimensional data could be transformed into low-dimensional codes using a multilayer neural network. From then on, CNNs have been used in numerous fields such as medical, security, forestry, and gained ongoing attention in both literature and business [Krizhevsky, Sutskever and Hinton (2012); He, Zhang, Ren et al. (2016); Lin, Chen and Yan (2014)]. Because deep learning has strong advantages in image recognition, this document makes use of deep learning to recognize food-ingredient images. Notably, ResNet beat other CNNs

including VGG, GoogLeNet, and gained the best scores on the ILSVRC (ImageNet Large Scale Visual Recognition Competition) 2015 recognition work. The depth and width of CNNs are extended rapidly, that means the more high-level and richer features are available using deep networks [Pouyanfar, Chen and Shyu (2017)].

One important issue is that CNNs need a large-scale image dataset to train a CNN module, while a small-scale dataset cannot be trained on CNNs because of overfitting. So far, two important methods have been applied to resolve the problem. One skill is fine-tuning that utilizes an already trained module, adjusts the CNN's framework, and restarts training from the module [Yanai and Kawano (2015)]. Another solving technique is using a pre-trained CNN module with a large-scale dataset as a deep-feature extractor of a small-scale dataset. The approach [Chen, Xu, Xiao et al. (2017)] applied a trained deep learning model to detect different types of food ingredients, and its best accuracy is close to 60%. Another problem is whether high-dimension feature vectors from a pre-trained CNN model on a different dataset (e.g., ImageNet) enhances accuracy of food-ingredient recognition. Several kinds of literature have demonstrated the usefulness of deep features for image detection [Pan, Pouyanfar, Chen et al. (2017); Yanai and Kawano (2015); Zhang, Isola, Efros et al. (2018)].

To resolve the aforementioned problems, this report presents an image augmentation-based food recognition technique for small and medium-scale datasets with CNNs. The new method utilizes image transformation and pretrained CNN models to overcome the problem of small dataset limitation, extracts high-level and valid image features using deep learning, and recognizes food ingredients. The extensive experimental results prove that the presented image augmentation-based food recognition architecture outstandingly promotes food detection accuracy compared to the existing methods.

The rest of this study is organized as follows. Section 2 introduces an overview of the state-of-the-art research in food recognition and CNNs. The details of the presented food recognition framework based on image augmentation are described in Section 3. Section 4 analyzes the experimental results on different image augmentation datasets, deep learning benchmarks with F1-measure accuracy and time cost of food recognition based on various deep-feature sets. Finally, Section 5 provides the concluding remark of the whole report.

2 Related work

This document will describe the relevant research including food detection and Convolutional Neural Networks as follows.

2.1 Food classification

Recently, food classification gained rapid development in machine learning. Such as: He et al. [He, Xu, Khanna et al. (2014)] and Nguyen et al. [Nguyen, Zong, Ogunbona et al. (2014)] extracted both local and global features for food detection. The former used the k-nearest neighbors and vocabulary trees, while the latter combined the partial figure and structural characteristics of food contents for food recognition. In paper Farinella et al. [Farinella, Moltisanti and Battiato (2014)], visual word distributions (Bag of Textons)

was regarded as food images and a Supported Vector Machine (SVM) was used to detect them. In document Bettadapura et al. [Bettadapura, Thomaz, Parnami et al. (2015)], the context of where a food image was exploited to represent food features for food-meal recognition. These food images were comprised of actually existing foods that were labeled as follows: American, Indian, Italian, Mexican, and Thai. A Japanese food dataset was made use of food classification on paper Joutou et al. [Joutou and Yanai (2009)]. This literature presented a multiple kernel learning method that mixed different image features including color, texture, and Scale Invariant Feature Transform (SIFT), and the food dataset which was composed of 50 categories of manually collected pictures from the Internet. Hoashi et al. [Hoashi, Joutou and Yanai (2010)] applied several kernel learning for feature fusion, and obtained 62.5% accuracy rate for image classification based on a dataset composed of 85 kinds of food pictures. The Pittsburgh Fast-food Image Dataset (PFID) [Chen, Dhingra, Wu et al. (2009)] involved 101 kinds of foods and three pictures for each class, which was the first open food dataset. Chen et al. [Chen, Yang, Ho et al. (2012)] used a food dataset composed of 50 kinds of Chinese foods. Another food recognition technique was presented with picking up dissimilar parts with Random Forest, and evaluated on the Food-101 dataset (downloaded from foodspotting.com) which obtained 50.76% average accuracy.

Currently, CNNs has been extremely valid for large-scale image classification and applied to food detection. A rapid auto-clean deep learning model was presented for food recognition [Chen, Xu, Xiao et al. (2017)]. This arcticle constructed a fine-tuning technology using deep learning for food recognition. Another DeepFood framework [Pan, Pouyanfar, Chen et al. (2017)] was proposed that used deep learning to extract deep features and selected deep feature sets with Information Gain selector. The architecture improved the classification accuracy. Kagaya et al. [Kagaya, Aizawa and Ogawa (2014)] leveraged deep learning for food classification with a dataset including ten kinds of foods from an open food-logging program. Kagaya et al. [Kagaya and Aizawa (2015)] recognized food/non-food pictures using deep learning on three datasets. A deep-learning food classification was presented utilizing both a patch-wise manner and a voting technique with a six-layer CNN [Christodoulidis, Anthimopoulos and Mougiakakou (2015)]. Ciocca et al. [Ciocca, Napoletano and Schettini (2017)] proposed a food recognition algorithm on an UNIMIB2016 food dataset including 73 food categories and a whole of 3616 food images. This work applied several features to detect food, and their experimental conclusion proved that the deep-learning features got a higher classification accuracy.

2.2 Convolutional neural networks

Deep learning is making unbelievable improvements in computer vision, speech recognition, natural language processing, and so on. Significantly, CNNs are exploited for computer vision, and deep convolutional neural networks have attained eminent advancements in image recognition [Krizhevsky, Sutskever and Hinton (2012); He, Zhang, Ren et al. (2016)].

AlexNet [Krizhevsky, Sutskever and Hinton (2012)] is the first framework using deep convolutional layers for image recognition. The architecture has eight layers including five convolutional layers and three fully connected layers, which contains multiple

convolutional and pooling layers put on top of each other rather than an individual convolutional layer followed by a pooling layer. In ILSVRC (ImageNet Large Scale Visual Recognition Challenge) 2012, AlexNet remarkably achieved better performance than the other high-ranking techniques.

Nowadays, Deep Residual Learning [He, Zhang, Ren et al. (2016)] acts as the benchmark of CNNs. Residual Network (ResNet) created by He et al. [He, Zhang, Ren et al. (2016)] from Microsoft who gained the champion of ILSVRC 2015 and COCO (Common Objects in Context) 2015 competitions on ImageNet recognition and localizations, as well as COCO segmentation and recognition. The CNN's outstanding accomplishment is the reconstructed learning process that directs the deep neural network information flow and decreases the degradation. ResNet is extremely deeper than other CNNs, and the residual framework has been demonstrated to construct a deeper CNN than before comfortably.

Recently, a novel CNN called "DenseNet" [Huang, Liu, Maaten et al. (2017)] was designed with dense connections. In DenseNet, connection of each layer utilizes a feed-forward fashion. Especially, DenseNet encourages feature reuse using the connection of features on the channel.

All the above mentioned deep learning frameworks, including other popular CNNs, have brought about numerous advancements in computer vision. As we all know, large-scale datasets are necessary for training a deep learning model. However, a large-scale dataset means that a large number of images and diversities of objects, which is not easy to obtain, while small datasets are very widespread and easy to be collected. Consequently, this document proposes an image augmentation-based food recognition architecture for small and medium-scale food datasets with CNNs.

3 The image augmentation-based food recognition framework

This report proposes a novel architecture of food-ingredient recognition utilizing image augmentation and CNNs. The framework is depicted in Fig. 3, which is composed of three major modules: (1) Image augmentation using rotation and flipping, (2) the last pooling-layer feature extraction using ResNet, (3) classification with SMO (Sequential Minimal Optimization).

3.1 Image augmentation

A CNN has numerous parameters that need to be trained, and the number of images is a key factor of deep learning using CNNs because the small datasets easily result in overfitting. A normal approach is image augmentation that artificially enlarges the size of a dataset [Krizhevsky, Sutskever and Hinton (2012)]. Classic augmentation techniques on images have affine transformations including translation, rotation, scaling, flipping, to name a few [Roth, Lu, Liu et al. (2016)]. In order to both enlarge the size of the food-ingredient dataset and preserve food characteristics, the framework utilizes both rotation and flipping to augment food images.

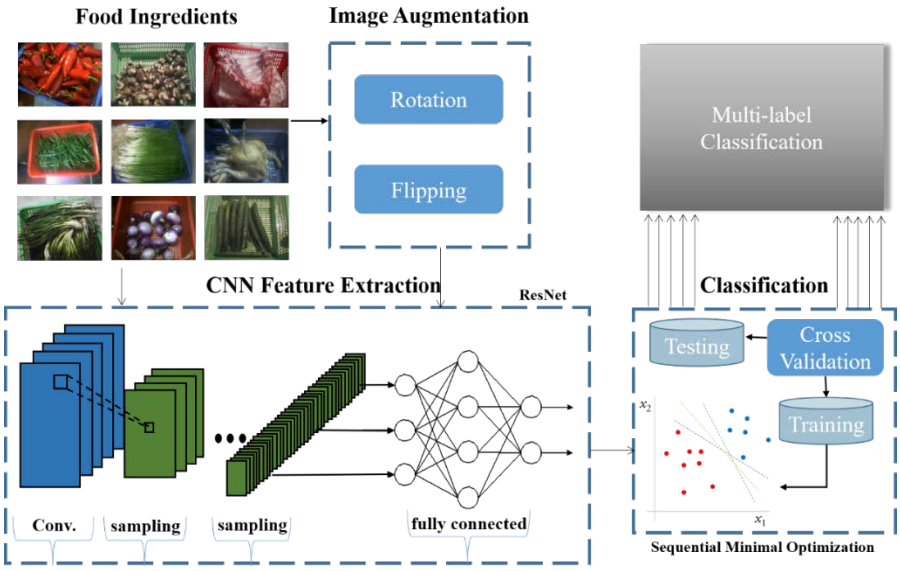

Figure 3: The image augmentation-based food recognition framework

Figure 4: An original food image and its augmentation with rotation and flipping

Firstly, each image is flipped N_f times (Vertical, Horizontal, and Horizontal & Vertical). Then, the original and flipped food images are rotated N_r times at random angles $\psi = [-90°, \ldots\ldots, 90°]$. After the image-augmentation process, the size of food dataset at least will be scaled ($1+N_f+N_r$) times of the original size, even $\left(1 + N_f + \left(1 + N_f\right) \times N_r\right)$ times. A food image and its augmentations are shown in Fig. 4.

Algorithm 1 shows the image augmentation for food-ingredient images. During the image-augmentation process, the original food pictures are rotated a flipped so that one larger food-ingredient dataset will be gained. In our architecture, the original food pictures are defined $P=\{(p_i), i=1, 2, \ldots, N_p\}$, where p_i is denoted as the i^{th} image and N_p is

the number of the original food images. Flipping involves three ways denoted as FW={Vertical, Horizontal, and Horizontal & Vertical}, and rotation is defined as $RW=\{(rw_k), k=1, 2, \ldots, rw_{Nr}\}$, where rw_k is the k's rotation angle. Line 2 to Line 7 show that each image in P will be rotated and flipped, and both the rotated food dataset RP and the flipped food set FP are output in Line 9.

Algorithm 1. the image augmentation for food ingredients

Input: Food images $P=\{(p_i), i=1, 2, \ldots, N_p\}$, flipping FW={Vertical, Horizontal, and Horizontal & Vertical}, and rotation $RW=\{(rw_k), k=1, 2, \ldots, rw_{Nr}\}$ that rw_k is the rotation angle.

Output: Rotated food images RP and flipped food images FP

1: **for** each image $p_i \in P$ do
2: **for** k=1 to N_r do
3: Rotating at the angle rw_k
4: **end for**
5: **for** FW ={Vertical, Horizontal, and Horizontal & Vertical}
6: Flipping
7: **end for**
8: **end for**
9: **return** two food datasets both *RP and FP*

3.2 The last pooling layer for feature extraction using ResNet

Recognizing a small-size image dataset is universal in the real world, while training a CNN model using a small-size dataset is impossible from scratch owing to overfitting. Alternatively, transfer learning is a well-liked method for recognizing medium and small-scale datasets. In deep learning field, transfer learning is the procedure of utilizing a pre-trained deep learning model such as a CNN model which is initially trained on a large-scale dataset (e.g., ImageNet) and acted as a fixed feature extractor for any size of datasets, including small or medium sets. The original pictures are granted as the input of a pre-trained CNN model and then CNN vectors are attained from its middle layers. The activation vectors are spread into the upper layers and the produced high-level vectors can be treated as the image description. Generally, the image deep features are extracted from the last output layers of the pre-trained deep model. On document [Pan, Pouyanfar, Chen et al. (2017)], experimental results showed that the second last layer of the pre-trained CNN had better performance than the last layer for food-ingredient classification. Therefore, our framework uses the last pooling layer of a pre-trained ResNet model to extract deep features.

A CNN is a multilayer artificial neural network that combines both unsupervised feature extraction and image recognition. In Fig. 3, the high-level features are extracted from the last pooling layer of RestNet. ResNet [He, Zhang, Ren et al. (2016)] is an extremely powerful CNN and shows superior recognition compared to other CNNs. It contains amazing residual connections and widely exploits batch normalization. Till

now, Resnet becomes a milestone of CNNs and brings superior improvements on visual image applications.

In Fig. 3, plenty of features will be generated when local areas of the whole input are iteratively operated with a function in a convolutional layer. As shown in Eq. (1), the k^{th} CNN vector at the k^{th} layer is noted as γ_{ij}^{k} where k is the assigned layer, i and j are dimensions of the input data, γ_{ij}^{k-1} is input data of the k^{th} layer from output of the k-1 layer, λ is an activation function such as Relu, and the filters of the k^{th} layer are defined as ω_{ij}^{k} (weights) and θ_{j}^{k} (bias). A pooling layer is a nonlinear down-sampling following by each convolutional layer. The pooling layer takes a small part from the preceding convolutional layer and produces an individual vector as depicted in Eq. (2), where σ_{ij}^{k} is a multiplicative bias and down(.) is a subsampling function like average pooling and max pooling.

$$\gamma_{ij}^{k} = \lambda\left(\omega_{ij}^{k} \times \gamma_{ij}^{k-1}\right) + \theta_{j}^{k} \tag{1}$$

$$\gamma_{ij}^{k} = \lambda\left[\sigma_{ij}^{k} down\left(\gamma_{ij}^{k-1}\right) + \theta_{j}^{k}\right] \tag{2}$$

In our architecture, the deep-feature extraction benefits from transfer learning using ResNet. First, the dataset is divided into training set T and testing set T'. T is denoted as $T=\{(t_1, c_q), (t_2, c_q), \ldots, (t_N, c_q)\}$, where t_i is the i^{th} training sample, N is the size of training samples, and $c_q \in C$ is the one certain category of food ingredients, where q is less than Nc. Classes $C=\{c_1, c_2, \ldots, c_{Nc}\}$ and Nc is the total kinds of food ingredients. Second, the pre-trained ResNet model and its last pooling layer are used as a deep feature extractor for unsupervised features. In addition, the extracted feature vectors are stored in $F=\{f_1, f_2, \ldots, f_{Ns}\}$, where f_i is the i^{th} feature vector from the last pooling layer, and N_s is the number of extracted feature vectors from the last pooling layer of ResNet.

In Fig. 3, utilizing the presented ResNet feature extractor will generate high-level, prosperous and valid deep features of food ingredients.

3.3 Image recognition

How to train an excellent recognized model and detect food images is a key problem when feature vectors are extracted. For the goal, our architecture uses SMO to train classification models (shown in Fig. 3). SMO is an ameliorated algorithm of Support Vector Machines (SVM) on detection assignments. It is constructed to get a valid solution of the expensive Quadratic Programming (QP) issue by splitting it into smallest probable sub-problems [Platt (1998)].

The recognition component includes two major procedures: training and testing. First, the image dataset is divided into training T and testing T' using three-fold cross validation. T has been denoted in Section III (2). $T' = \{(t_1, c_x), (t_2, c_x), \ldots, (t_{Nt}, c_x)\}$, where t_i is the i^{th} testing samples, Nt is the size of testing images, and $c_x \in C$ but the c_x is an unlabeled category. For the testing set, the t_i is a testing instance with unknown food type.

In the training process, several SMOs are trained for food-ingredient recognition using the training dataset T and the deep features F extracted during the feature extraction phase depicted in Section III (2). During the testing stage, the category of each testing

sample is predicted using the trained SMO models as shown in Algorithm 2. The inputs of the testing algorithm are composed of testing instances T' and its feature vectors F, as well as all the trained models *SMOs*. Its outputs are a predicted food category set *PL*, and an accuracy array *Acc*. In Algorithm 2, the accuracy is computed for each trained SMO model. The j^{th} testing sample is forecasted as PC_{ij} exploiting the i^{th} trained model SMO_i in Line 1. The testing samples whose types correctly predicted using SMO_i are amounted as PC_i in Line 5. Then, the accuracy of each trained SMO_i is counted as the corresponding Acc_i in Line 6. Finally, all of the predicted testing-instance classes and the average accuracies are output in Line 8.

Algorithm 2. the predicted classes of testing samples

Input: Testing instances $T' = \{(t_i, c), i = 1,2, \dots, N_t\}$ and the last pooling layer features $F=\{f_1, f_2, \dots, f_{Ns}\}$, Trained models *SMOs*

Output: Predicted each instance's class PC_{ij} and average accuracy Acc_i

1: **for** each trained model $SMO_i \in SMOs$ do
2: **for** each test instance $(t_j, c) \in T'$
3: $PC_{ij} \leftarrow SMO_i((t_j, c))$
4: **end for**
5: $PC_i \leftarrow \sum$ Correctly recognized samples
6: $Acc_i = PC_i / Nt$
7: **end for**
8: **return** PC_{ij}, Acc_i

4 Experimental analysis

4.1 Food-ingredient dataset

The study involves the MLC-41 dataset [Pan, Pouyanfar, Chen et al. (2017)] which is a small-scale set of food-ingredient images originated from a large food supply chain platform in China (Mealcome dataset) [Chen, Xu, Xiao et al. (2017)]. The raw food-ingredient images were gathered in a sophisticated scene which mixed different backgrounds and food ingredients. Most of the initial images are clear to be distinguished by the human eye, while some are hard to be detected as the labeled food-ingredient categories because of blurriness, noise, illumination, overexposure, or some other reasons. Consequently, the noisy images were removed and obviously recognized images were reserved and labeled into the corresponding food-ingredient categories. Finally, a small-scale food dataset is constructed called the MLC-41 dataset which contains forty-one kinds of food ingredients and each category includes one hundred pictures, and each picture resolution is adjusted to 640*480 pixels to get a more efficient feature extraction and food-ingredient classification. MLC-41 dataset is a balanced set, but the size of categories is a bit high compared with the size of pictures in each category. This makes the training task more challenging. Fig. 5 shows several instances of the MLC-41 dataset as below, like Carrot, Red Pepper, Cabbage, to name a few.

4.2 Experimental setup

For image recognition, how to evaluate our proposed architecture is very significant. Generally, evaluation metrics like F1, Precision, and Recall are appropriate for 0-1 detection, particularly imbalanced data. The MLC-41 dataset is a balanced set and the recognition task is multiclass detection. Consequently, the accuracy metric is exploited to evaluate the image-augmentation framework presented on this study.

Normal affine transformations include translation, rotation, shearing, flipping, and so on. In order to evaluate our presented framework, translation and shearing are utilized for image augmentation. Caffe [Jia, Shelhamer, Donahue et al. (2014)] is a common deep learning platform which was created by Yangqing Jia, evolved by Berkeley AI Research (BAIR) and community contributors. Caffe involves plentiful pre-trained CNN models including AlexNet, CaffeNet [Jia, Shelhamer, Donahue et al. (2014)], and ResNet-50 and so on. In experiments, the novel presented feature extractor is compared with AlexNet and CaffeNet models. In this work, feature vectors are extracted from the second last layer of each CNN model. For example, the second last layer of CaffeNet and AlexNet is the layer “fc7”, which produces a 4096-dimension feature vector, and of ResNet-50 is the “pool5” which outputs a 2048-dimension feature vector. Additionally, the average accuracy of the image-augmentation dataset is compared with that of various food datasets utilizing three-fold cross validation.

Figure 5: Image samples of MLC-41dataset

4.3 Experimental results

The framework based on image augmentation for food-ingredient classification is analyzed on the MLC-41 dataset. This experiment utilizes various affine translations to augment images, such as flipping, rotation, translation. The second last layer of CNNs is exploited as a deep-feature extractor. The SMO classifier is adjusted to achieve to its best capability on all evaluated food datasets, and measured with the three-fold cross validation.

This experiment uses classic augmentation techniques including rotation, flipping, translation and shearing. Tab. 1 shows the sizes of original and different image-

augmentation datasets. In Tab. 1, the original food dataset has 4100 images. All image-augmentation datasets are built based on affine transformation of the original dataset. The Rot. Dataset uses rotation and flipping and it is five times that of the original dataset. The Tra. dataset uses translation and shearing and includes 4100*8 images. The Rot. & Tra. dataset has 4100*13 pictures combined with the Rot. and Tra. Datasets. The Tra. & Ori. dataset includes 4100*9 images and that is the combination of Tra. and Ori. datasets. The Rot. & Tra. & Ori. dataset is fourteen times that of the original set because it combines the three datasets of Rot., Tra. and Ori. The dataset of Rot. & Ori has 4100*6 images.

Table 1: Different sizes of the original and image-augmentation datasets

Dataset	**Ori.**	**Rot.**	**Tra.**	**Rot. & Tra.**	**Tra. & Ori.**	**Rot. & Tra. & Ori.**	**Rot. & Ori.**
Size	4100	4100*5	4100*8	4100*13	4100*9	4100*14	4100*6

Table 2: Average accuracy difference between various CNN models and datasets

Deep Learning Model	**Cross Val.**	**Ori.**	**Rot.**	**Tra.**	**Rot. & Tra.**	**Tra. & Ori.**	**Rot. & Tra. & Ori.**	**Rot. & Ori.**
AlexNet	Fold1	79.84	80.13	76.26	79.48	77.91	79.34	80.56
	Fold2	81.08	79.97	77.90	79.97	79.38	80.49	81.08
	Fold3	79.75	80.41	79.75	81.30	80.41	81.37	81.60
	Avg.	**80.22**	**80.17**	**77.97**	**80.25**	**79.23**	**80.40**	**81.08**
CaffeNet	Fold1	78.55	79.41	76.83	79.05	78.26	79.27	79.12
	Fold2	81.60	81.61	74.65	81.67	80.86	81.37	81.52
	Fold3	80.86	80.78	79.45	80.78	80.56	81.08	81.15
	Avg.	**80.33**	**80.60**	**76.98**	**80.50**	**79.89**	**80.57**	**80.60**
ResNet	Fold1	86.15	87.37	85.15	87.23	86.30	87.37	88.09
	Fold2	88.40	89.43	87.80	88.40	88.17	88.25	89.43
	Fold	87.95	88.25	86.84	89.36	88.10	89.36	88.99
	Avg.	**87.63**	**88.35**	**86.60**	**88.33**	**87.52**	**88.33**	**88.84**

The average accuracy of different image augmentation datasets integrated with various CNN modules are shown in Tab. 2. From Tab. 2, the deep features extracted with CNNs from different food datasets promote the detection accuracy of the 41-kinds of food ingredients. Specifically, the combination food dataset of Rot. & Ori. beats other sets including original, translation and other combination datasets when their features are extracted from the last pooling layer of ResNet. The average accuracy corresponding to the food dataset constructed by the image-augmentation framework with RestNet reaches the highest, where the average accuracy of 41-kinds of food recognition gains 88.84%.

From Tab. 2, we observe that the deep features extracted from the Ori. dataset keep a

better average accuracy than the translation deep features. The average performance of the Rot. dataset is extremely near to the Ori. dataset. However, using our presented framework with the combination of Rot. & Ori. datasets gains the most outstanding deep-feature vectors and reaches the best performance utilizing ResNet. From Tab. 2, the accuracy of Rot. & Tra. & Ori. dataset is not better than the Rot. & Ori. set in classification. It means that more deep features don't indicate the better average accuracy of food classification. Consequently, the experiment results prove that the combinational dataset of rotation, flipping, and original is more effective to extract image features than other food sets including original datasets. The main reason is referred that rotation and flipping techniques both enrich the food-ingredient images, and preserve the original image characteristics.

In the image-augmentation architecture, deep features are extracted from the last pooling layer of ResNet. From Tab. 2, it can be noted that ResNet achieves much better recognition accuracy (nearly 10%) than other CNNs. The almost identical difference has also produced in Tab. 3. Therefore, it is proved again that ResNet defeats AlexNet and CaffeNet for food-ingredient recognition.

Table 3: Average accuracy difference between various CNNs and classifiers

Model \ Classifier	Random Forest	Bagging	Multi-Class Classifier Updateable	BayesNet	SMO
AlexNet	70.14	58.61	68.76	68.17	81.08
CaffeNet	69.37	57.59	68.89	67.66	80.60
ResNet	81.72	72.82	80.75	76.81	**88.84**

Tab. 3 shows the average accuracy difference between various CNN models and classifiers using the Rot. & Ori. dataset. These experimental results reveal that the image-augmentation architecture is advanced to other approaches for multi-category classification of food ingredients, which combines the dataset of Rot. & Ori., ResNet deep learning, SMO recognition. As we can see from Tab. 3, the highest recognition effects have the average accuracy of 81.08%, 80.60%, and 88.84%, which are totally generated using the framework both SMO and deep-feature vectors. The another better average accuracy reach to 70.14%, 69.37% and 81.72% utilizing the Random Forest classifier. From Tab. 2 and Tab. 3, a conclusion can be inferred that the image-augmentation architecture which obviously promotes the correctness of the food recognition. The best architecture is obtained when combining the image augmentation both rotation and flipping, ResNet deep feature dataset, and SMO classifier, which beats other techniques and gets the best accuracy for multi-class recognition of the MLC-41 dataset.

Tab. 4 lists time cost used to build SMO models with different sizes of food sets. From Tab. 4, we can see that the building time is longer when the size of the dataset is larger. This major reason is that a larger dataset including more images needs more time to build the SMO classifier models. Tab. 5 shows the testing time on the same size of food-ingredient datasets with the trained SMO models. It takes only 3.32 seconds with the trained SMO model on the Rot. & Ori. dataset and ResNet, which is a little longer than

3.15 seconds with the trained SMO model on the Tran. dataset and ResNet. This shows our proposed framework is very efficient and it reduces the classifying time of SMO models on the same dataset.

Table 4: Time (Seconds) taken to build SMO models with datasets

Deep Learning Model	Cross Val.	Ori.	Rot.	Tra.	Rot. & Tra.	Tra. & Ori.	Rot. & Tra. & Ori.	Rot. & Ori.
AlexNet	# 1	23.22	194.84	574.30	1046.35	648.11	1127.06	264.95
	# 2	23.42	205.77	598.77	1112.36	669.97	1209.45	254.84
	# 3	23.00	206.44	607.75	1124.43	688.39	1211.66	255.90
	Avg.	**23.21**	**202.35**	**593.61**	**1094.38**	**668.82**	**1182.72**	**258.56**
CaffeNet	# 1	23.05	195.06	552.26	996.77	633.31	1122.28	238.32
	# 2	26.33	205.66	190.07	550.04	668.09	909.04	249.61
	# 3	23.46	204.75	596.33	1072.89	659.62	1191.08	249.76
	Avg.	**24.28**	**201.82**	**446.22**	**873.23**	**653.67**	**1074.13**	**245.90**
ResNet	# 1	10.93	77.73	158.25	282.70	177.08	299.89	101.43
	# 2	12.53	79.73	163.76	13.33	184.28	316.68	98.03
	# 3	14.90	80.01	164.84	324.59	184.52	319.49	101.67
	Avg.	**12.79**	**79.16**	**162.28**	**206.87**	**181.96**	**312.02**	**100.38**

Table 5: Time (Seconds) taken to test SMO models on the same size of food sets

Deep Learning Model	Cross Val.	Ori.	Rot.	Tra.	Rot. & Tra.	Tra. & Ori.	Rot. & Tra. & Ori.	Rot. & Ori.
AlexNet	# 1	7.79	6.59	6.71	8.36	6.86	10.71	6.57
	# 2	7.51	6.99	6.70	10.69	10.04	6.57	6.87
	# 3	11.02	7.27	6.73	7.71	5.86	5.90	7.07
	Avg.	**8.77**	**6.95**	**6.71**	**8.92**	**7.59**	**7.73**	**6.84**
CaffeNet	# 1	10.47	11.06	28.63	28.03	7.93	10.99	6.84
	# 2	7.51	6.64	26.71	27.75	7.56	7.66	10.01
	# 3	13.47	6.43	29.18	28.03	7.79	7.63	7.76
	Avg.	**10.48**	**8.04**	**28.17**	**27.94**	**7.76**	**8.76**	**8.20**
ResNet	# 1	4.40	5.62	3.45	5.74	5.59	3.08	3.58
	# 2	3.80	3.19	3.02	3.33	3.26	2.94	2.97
	# 3	4.04	3.27	2.98	3.48	3.78	5.47	3.40
	Avg.	**4.08**	**4.03**	**3.15**	**4.18**	**4.21**	**3.83**	**3.32**

To further measure the presented image-augmentation architecture, compared with another two works using the similar food-ingredient dataset [Chen, Xu, Xiao et al. (2017); Pan, Pouyanfar, Chen et al. (2017)]. The method [Chen, Xu, Xiao et al. (2017)], the top1 accuracy with CaffeNet was below 50% and close to 60% with AlexNet, while the average accuracy of the novel image-augmentation architecture with CaffeNet is 80.60%, and with AlexNet is 81.08% as shown in Tab. 6. Tab. 6 depicts the average accuracy difference between two food recognition techniques with the MLC-41 dataset and various CNNs. As can be noted from Tab. 6, the image-augmentation architecture is close to the DeepFood framework [Pan, Pouyanfar, Chen et al. (2017)] using AlexNet and CaffeNet. Significantly, the method presented in this document achieves the best average accuracy with image augmentation and ResNet than other benchmarks, and the average accuracy attains 88.84%. It is an important promotion compared to the approach [Chen, Xu, Xiao et al. (2017)] and superior to the DeepFood framework.

Table 6: Accuracy comparison between two frameworks

Framework / Model	DeepFood	Image Augmentation
AlexNet	80.42	81.08
CaffeNet	80.76	80.60
ResNet	87.78	**88.84**

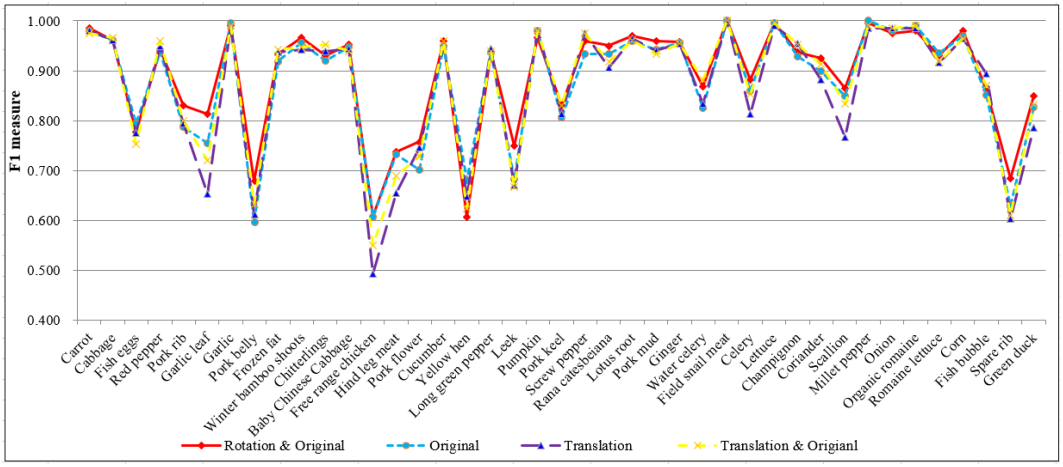

Figure 6: F1 measure for 41-class food ingredients on different datasets

Fig. 6 depicts the graphic average accuracy difference of various food-ingredient datasets. In Fig. 6, the F1 Measure of each food category is marked. As can be noticed from this figure, the combinational dataset of Rotation & Original outperforms other food datasets in all food types except Yellow hen, and the corresponding F1 values of several food categories, are obviously higher than other datasets. Overall, the F1 Measure plot of each dataset on forty-one categories is approximately fluctuating from 70% to 100%, and the

accuracy of several classes gain or come near to 100%. Therefore, the novel presented architecture strongly strengthens the effectiveness of food recognition.

In sum, it can be concluded that the novel Image-augmentation architecture integrates the advantages of the image augmentation with affine transformations, deep feature extraction using ResNet and SMO classifier, and achieves very high effectiveness for food recognition comparing with earlier techniques. Furthermore, the proposed architecture promotes the image recognition using CNNs for small-scale or medium datasets.

5 Conclusion

This literature proposes a novel approach, an image augmentation-based food recognition utilizing CNNs, which combines image augmentation and high-level feature vectors as well as SMO classifier. The new framework is designed for the classification of small or medium-scale datasets that is an extremely common and important assignment in real life. Therefore, it is applied to the image recognition of MLC-41 food ingredients. The Image-augmentation technique is measured with comprehensive experiments by comparing the average accuracy of various image transformation datasets, CNN models and classifiers. The extensive experimental results demonstrate the promotion and enhancement of the Image-augmentation architecture for food recognition. We believe that other classification problems for small or medium datasets can benefit from the Image-augmentation framework, and the presented method will lead to stronger classification systems.

Acknowledgement: The authors would like to acknowledge the financial support from the Key Research & Development Plan of Hunan Province (Grant No. 2018NK2012), Graduate Education and Teaching Reform Project of Central South University of Forestry and Technology (Grant No. 2018JG005), and Teaching Reform Project of Central South University of Forestry and Technology (Grant No. 20180682).

References

Bettadapura, V.; Thomaz, E.; Parnami, A.; Abowd, G. D.; Essa, I. (2015): Leveraging context to support automated food recognition in restaurants. *IEEE Winter Conference on Applications of Computer Vision*, pp. 580-587.

Bossard, L.; Guillaumin, M.; Gool, L. V. (2014): Food-101-Mining discriminative components with random forests. *European Conference on Computer Vision*, vol. 8694, pp. 446-461.

Chen, C.; Zhu, Q.; Lin, L.; Shyu, M. L. (2013): Web media semantic concept retrieval via tag removal and model fusion. *ACM Transactions on Intelligent Systems & Technology*, vol. 4, no. 4, pp. 1-22.

Chen, H.; Xu, J.; Xiao, G.; Wu, Q.; Zhang, S. et al. (2017): Fast auto-clean CNN model for online prediction of food materials. *Journal of Parallel and Distributed Computing*, vol. 117, pp. 218-227.

Chen, M.; Dhingra, K.; Wu, W.; Yang, L.; Sukthankar, R. et al. (2009): PFID: Pittsburgh fast-food image dataset. *IEEE International Conference on Image Processing*, pp. 289-292.

Chen, M. Y.; Yang, Y. H.; Ho, C. J.; Wang, S. H.; Liu, S. M. et al. (2012): Automatic Chinese food identification and quantity estimation. *SIGGRAPH Asia 2012 Technical Briefs*, pp. 1-4.

Christodoulidis, S.; Anthimopoulos, M.; Mougiakakou, S. (2015): Food recognition for dietary assessment using deep convolutional neural networks. *International Conference on Image Analysis and Processing*, pp. 458-465.

Ciocca, G.; Napoletano, P.; Schettini, R. (2017): Food recognition: a new dataset, experiments, and results. *IEEE Journal of Biomedical & Health Informatics*, vol. 21, no. 3, pp. 588-598.

Farinella, G. M.; Moltisanti, M.; Battiato, S. (2014): Classifying food images represented as bag of textons. *IEEE International Conference on Image Processing*, pp. 5212-5216.

He, K.; Zhang, X.; Ren, S.; Sun, J. (2016): Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778.

He, Y.; Xu, C.; Khanna, N.; Boushey, C. J.; Delp, E. J. (2014): Analysis of food images: features and classification. *IEEE International Conference on Image Processing*, pp. 2744-2748.

Hinton, G. E.; Salakhutdinov, R. R. (2006): Reducing the dimensionality of data with neural networks. *Science*, vol. 313, no. 5786, pp. 504-507.

Hoashi, H.; Joutou, T.; Yanai, K. (2010): Image recognition of 85 food categories by feature fusion. *IEEE International Symposium on Multimedia*, pp. 296-301.

Huang, G.; Liu, Z.; Maaten, L.; Weinberger, K. Q. (2017): Densely connected convolutional networks. *Computer Vision and Pattern Recognition*.

Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J. et al. (2014): Caffe: convolutional architecture for fast feature embedding. *Proceedings of the 22nd ACM International Conference on Multimedia*, pp. 675-678.

Joutou, T.; Yanai, K. (2009): A food image recognition system with multiple kernel learning. *IEEE International Conference on Image Processing*, pp. 285-288.

Kagaya, H.; Aizawa, K. (2015): Highly accurate food/non-food image classification based on a deep convolutional neural network. *International Conference on Image Analysis and Processing*, pp. 350-357.

Kagaya, H.; Aizawa, K.; Ogawa, M. (2014): Food detection and recognition using convolutional neural network. *ACM International Conference on Multimedia*, pp. 1085-1088.

Krizhevsky, A.; Sutskever, I.; Hinton, G. E. (2012): ImageNet classification with deep convolutional neural networks. *International Conference on Neural Information Processing Systems*, pp. 1106-1114.

Li, H.; Qin, J.; Xiang, X.; Pan, L.; Ma, W. et al. (2018): An efficient image matching algorithm based on adaptive threshold and RANSAC. *IEEE ACCESS*, vol. 6, pp. 66963-66971.

Lin, M.; Chen, Q.; Yan, S. (2014): Network in network. *International Conference on Learning Representations.*

Liu, W.; Wang, Z.; Liu, X.; Zeng, N.; Liu, Y. et al. (2017): A survey of deep neural network architectures and their applications. *Neurocomputing*, vol. 234, pp. 11-26.

Nguyen, D. T.; Zong, Z.; Ogunbona, P. O.; Probst, Y.; Li, W. (2014): Food image classification using local appearance and global structural information. *Neurocomputing*, vol. 140, pp. 242-251.

Pan, L.; Pouyanfar, S.; Chen, H.; Qin, J.; Chen, S. C. (2017): DeepFood: automatic multi-class classification of food ingredients using deep learning. *IEEE International Conference on Collaboration and Internet Computing*, pp. 181-189.

Platt, J. (1998): A fast algorithm for training support vector machines. *Journal of Information Technology*, vol. 2, no. 5, pp. 1-28.

Pouyanfar, S.; Chen, S. C. (2016): Semantic concept detection using weighted discretization multiple correspondence analysis for disaster information management. *IEEE International Conference on Information Reuse and Integration*, pp. 556-564.

Pouyanfar, S.; Chen, S. C.; Shyu, M. L. (2017): An efficient deep residual-inception network for multimedia classification. *IEEE International Conference on Multimedia and Expo*, pp. 373-378.

Roth, H. R.; Lu, L.; Liu, J.; Yao, J.; Seff, A. et al. (2016): Improving computer-aided detection using convolutional neural networks and random view aggregation. *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1170-1181.

Simonyan, K.; Zisserman, A. (2014): Very deep convolutional networks for large-scale image recognition. *Proceedings of International Conference on Learning Representations*.

Yanai, K.; Kawano, Y. (2015): Food image recognition using deep convolutional network with pre-training and fine-tuning. *IEEE International Conference on Multimedia & Expo Workshops*, pp. 1-6.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; Wang, O. (2018): The unreasonable effectiveness of deep features as a perceptual metric. arXiv:1801.03924.