Hindawi
Mathematical Problems in Engineering
Volume 2020, Article ID 6253827, 8 pages
https://doi.org/10.1155/2020/6253827

Research Article

Enhanced Mask R-CNN for Chinese Food Image Detection

Y. Li, X. Xu, and C. Yuan

School of Mathematics and Computer Science, Wuhan Polytechnic University, Wuhan 430023, China

Correspondence should be addressed to C. Yuan; yc@whpu.edu.cn

Received 10 June 2020; Revised 8 July 2020; Accepted 16 July 2020; Published 30 July 2020

Guest Editor: Jun-Jun Jiang

Copyright © 2020 Y. Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Food image detection plays an essential role in visual object detection, considering its applicability in solutions that improve people's nutritional status and thus their health-care. At present, most food detection technologies are aimed at Western food and Japanese food, but few at Chinese foods. In this work, we exert effort to establish a Chinese food image dataset called CF-108 that can be used as an essential data basis for Chinese food image detection. The CF-108 dataset contains most Chinese dishes and covers large variations in presentations of the same category. In addition, we introduce a training architecture that replaces the traditional convolution in mask region convolutional neural network (Mask R-CNN) with depthwise separable convolution, namely, Mask R-DSCNN, to reduce the expensive computation cost. Experiments demonstrate that Mask R-DSCNN can significantly reduce resource consumption and improve Chinese food images' detection efficiency without hurting too much accuracy.

1. Introduction

Diet is essential for people's dietary health and quality of life [1]. By effectively recognizing and segmenting food images of daily meals, people can obtain practical information and effectively analyze and summarize their needs [2]. For example, ordinary people can balance their nutrition intake [3,4]; people with diabetes can avoid all high-sugar dishes [5]; doctors can also analyze the patient's previous diet structure and give reasonable dietary recommendations [6].

Food image recognition has always been a topic of concern. In [7], statistical methods were proposed to calculate the characteristics of dish images for food image classification. The author in [8] used the random forest to extract local features to classify dishes. In [9], the classification of dish images was discussed based on texture Anti-Textons features. In these cases, the performance of model accuracy and generalization ability is poor. With the development of deep learning, using a convolutional neural network to classify images occupies a dominant position and many other computer vision tasks [10–12]. Kagaya et al. [13] introduced the deep learning model Alex Net to detect and classify food images. Hassannejad et al. [14] showed a deeper model inception to classify food images. Singla et al. [15] have applied the GoogLeNet model to classify food images and non-food images. The development of object detection technology [16,17] has put forward higher requirements for image recognition. The main popular algorithms can be divided into two categories. One is based on Region Proposal's convolution neural network, called R-CNN [18], including R-CNN, Fast R-CNN [19], Faster R-CNN [20], and Mask R-CNN [21]. These R-CNN algorithms are two-stage algorithms that first generate the target candidate boxes and then predict the detection results. The other is a single-stage algorithm, such as YoLo [22–24] and SSD [25], which only uses convolutional neural network CNN to directly predict the category and location of different targets.

Generally, the detection accuracy of the two-stage algorithm is better than that of the single-stage algorithm; especially, the performance of Mask-RCNN outperformed existing single-model entries in each task in the 2016 COCO Challenge [26]. However, Mask-RCNN is computationally expensive and time-consuming, due to its relatively complex model structure. In this paper, we present the work of establishing a Chinese food image dataset as an essential data basis for Chinese food image detection. We further propose using depthwise separable convolution [27] instead of

traditional convolution to reduce the number of model parameters and operation costs.

The main contributions of this paper are as follows:

(1) We built a dataset of Chinese food images called CF-108 for the Chinese food detection task containing 100,800 images of 108 categories covering most Chinese dishes.

(2) We apply the depthwise separable convolution instead of traditional convolution to the standard Mask R-CNN model, namely, Mask R-DSCNN, to reduce model consumption.

(3) The experiment results on CF108 demonstrate that the Mask R-DSCNN can greatly reduce resource consumption and improve the detection efficiency of Chinese food images without hurting too much accuracy.

The rest of the paper is organized as follows. Section 2 briefly reviews the target detection algorithm Mask R-CNN as background knowledge. Section 3 provides the procedures for building a dataset of Chinese food images. In Section 4, the framework of Mask R-DSCNN is formally set out with experiments of Chinese food detection in Section 5. This work is concluded in Section 6.

2. Background

In recent years, breakthroughs have been made in target detection algorithms. Among all these algorithms, Mask R-CNN outperformed existing single-model entries in each task in the 2016 COCO Challenge [26]. The framework of Mask R-CNN mainly includes three parts: first, the backbone convolutional neural network (CNN) for feature extraction from the input image; second, the Region Proposal Network (RPN) [28] using anchors with different scales and aspect ratios sliding on the feature map to generate region proposals; third, the three branches in a parallel prediction network with two fully connection (FC) layers for bounding-box classification and regression, and a fully convolutional network (FCN) [29] for predicting object mask.

In principle, the backbone network could be any main models of deep neural networks, such as AlexNet [30], VGG [31], GoogLeNet [32,33], and ResNet [34]. In the Mask-RCNN model, ResNet (remove the last fully connected layer) is used as the backbone network to extract features, which can effectively reduce the difficulty of gradient disappearance and training degradation without increasing model parameters. ResNet contains five sets of convolutions. The underlying network can extract low-level features such as edges, while the upper network can extract the top-level features representing the target category. In order to make better use of the features of each level, Mask-RCNN extends the backbone network to a feature pyramid network (FPN), which uses the inherent layering and multiscale properties of convolutional neural networks to derive useful features for object detection.

The goal of RPNs is to predict a set of region proposals efficiently. To this end, a small network will slide over the feature map and generate multiple region proposals with multiple scalars and aspect ratios based on anchors. Two FC layers then follow this feature for box regression (reg) and box classification (cls). In RPNs training, the anchors with the largest intersection-over-Union (IoU) overlapping with the ground truth box are used as positive labels, and the anchors with IoU ratio below 0.3 are used as negative labels. The calculation of IoU is shown in

$$\mathrm{IoU} = \frac{\text{Detection Result} \cap \text{Ground Truth}}{\text{Detection Result} \cup \text{Ground Truth}}, \tag{1}$$

where Detection Result indicates the predicted box, and the Ground Truth indicates the ground truth box. RPNs will fine-tune the region proposals based on the obtained regression information and delete those region proposals that coincide with the image boundary. Finally, according to Non-Maximum Suppression (NMS) [35], about 2000 proposal regions per image will be left.

The region proposals generated from RPN require RoIAlign to adjust their dimension to meet the multibranch prediction networks. RoIAlign uses bilinear interpolation, instead of the rounding operation in RoIPool in Faster R-CNN, to extract the corresponding features of each region proposal on the feature map. The multibranch prediction network consists of FC layers for object detection, and FCN for masking. During the model training process, the loss function of the Mask R-CNN model for each proposal is shown infd2

$$L = L_{\mathrm{cls}} + L_{\mathrm{box}} + L_{\mathrm{mask}}, \tag{2}$$

where L_{cls} and L_{box}, respectively, represent classification loss and regression loss, and L_{mask} represents segmentation loss; the specific calculation formula of classification and regression loss is shown infd3

$$L_{\mathrm{cls}} + L_{\mathrm{box}} = \frac{1}{N_{\mathrm{cls}}} \sum_i L_{\mathrm{cls}}\left(p_i, p_i^*\right) + \lambda \frac{1}{N_{\mathrm{box}}} \sum_i L_{\mathrm{box}}\left(t_i, t_i^*\right), \tag{3}$$

where i represents the index of the anchor, p_i indicates the predicted probability of anchor i, t_i represents the four coordinate parameters of the box, and t_i^* represents the coordinate parameters of the ground-truth box corresponding to the positive anchor. If the anchor is positive, p_i^* is 1; otherwise, p_i^* is 0. By minimizing the loss function, the model is gradually optimized.

3. Chinese Food Image Dataset

3.1. Data Collection. The goal to build a dataset of Chinese food images needs to meet the following three aspects. First, the dataset needs to contain as many Chinese food images as possible, and each item needs to be represented with as many images as possible. Besides, in practical scenarios, the resolution of the dish images is varied by the camera taken, meaning that a dataset containing pictures of multiple resolutions can provide a more accurate representation of food. Therefore, the sources of Chinese food images are wanted to contain dishes with different resolutions to yield

better robustness of the model. Finally, both the image and the target object must be correctly labeled.

To meet these goals, we first gather the labels and images of Chinese dishes using publicly available images from the relevant Chinese food websites (http://www.meishichina.com; http://www.douguo.com) where most of the users post their Chinese dishes with tags. The ten most common food items from these websites are shown in Table 1. Web crawler technology [36] is used to obtain the labels and images of Chinese dishes since it can effectively obtain data on a topic within a specific time and a specific range on the website, such as fried shrimp, braised pork, and pickled fish. As a result, the images crawled achieve over 100,000 of 108 categories. Each dish has at least more than 100 images and a maximum of more than 1,000 images.

3.2. Data Preprocessing. Typically, the collected data is complex and may contain inappropriate images, unclear images, or complex noise images. Therefore, the next step is to clean [37], smooth, and label the data to improve the quality of the dataset. In this step, we first remove the images that are unclear or irrelevant. Then we use median filtering [38] to smooth image noise caused by other unrelated objects on the target object (such as background debris or image watermarks). In addition, we label both the image and the target object to achieve the complete segmentation of the target object from the background.

Image histogram is a common method for data cleaning. The specific operation is to convert the image into a histogram and then use the correlation coefficient method to find the similarity of the image. Images with similarity below the threshold will be cleared. The calculation formula of the correlation coefficient is defined as

$$r(x, y) = \frac{\mathrm{Cov}(x, y)}{\sqrt{\mathrm{Var}[x]\mathrm{Var}[y]}}, \tag{4}$$

where x, y are the histogram results of the two images, $\mathrm{Var}[x]$ is the covariance of x, $\mathrm{Var}[y]$ is the covariance of y, and $\mathrm{Cov}(x, y)$ is the covariance between x and y. The value of this formula ranged from −1 to 1. The larger the calculated result, the more similar the two images. In this paper, the method of histogram is used to clean the collected Chinese food images. Before using histogram, we manually remove some images that are irregularly large or small, which usually are irrelevant images. Then, we select a correct image for each dish category and then calculate the correlation coefficient between this correct image and the remaining images separately. If the correlation coefficient value of any image is less than 0.3, we consider this image irrelevant to the correct image and remove it.

The biggest difference between image and noise is the change of gray level. The visual obstacle of the image is formed by the huge change between the gray level of noise and the surrounding gray level. Therefore, an image smoothing method is generally used to eliminate noise by utilizing the nature of gray scale differences. Figure 1 is a comparison of spicy crayfish images before and after smoothing by median filtering (the left side is before processing and the right side is after processing). In the image of the spicy crayfish on the right below, we can clearly see that the background debris, such as green onions, peppers, tea cups, and chopsticks on the table, lose many obvious bright spots, and the image becomes smoother.

Table 1: The 10 most frequent dishes from the relevant websites of Chinese food.

Item	Quantity
Braised pork	665
Shredded cabbage	457
Braised fish	564
Dried green beans	746
Tomato egg soup	586
Fried shrimp	457
Iron plate beef	467
Noodles with sesame paste	435
Sweet and sour pork ribs	387
Scrambled eggs with pepper	698

After data cleaning and data smoothing, the next step is data labeling. Labeling Chinese dishes in dataset building is an expensive process because, even in the same category, the food images appear considerably different in various ingredients and cooking styles. In this experiment, we adopt the same labeling method used in [39], which designs a semisupervised method to accelerate the labeling process. Specifically, it pretrains a CNN model for the food recognition task based on some labeled samples and then classifies the collected images into candidate labels according to this CNN model. Finally, the label images are completed by manually performing label verification to finalize the dataset. Note that both image and target object need to be labeled to achieve the complete segmentation of the target object from the background.

3.3. Dataset Description. After work of data collection and data preprocessing, finally, the new Chinese food image dataset CF-108 contains 100,800 images of 108 categories, each of which covers significant variations in presentations of the same category. We divide the dataset into training and testing sets approximately at a ratio of 8 : 2. Specifically, there are 81,543 and 19,257 images for training and testing sets, respectively. Figure 2 shows some example images with their original size in the CF-108 dataset.

4. Mask R-DSCNN

4.1. Depthwise Separable Convolution. Depthwise separable convolution, proposed by Laurent Sifre in 2013 [27], has the characteristics of lower parameter quantity and operation cost compared with the standard convolution operation [40]. The main idea of depthwise separable convolution is to decompose the standard convolution integral into depthwise convolution and pointwise convolution. The comparison between depthwise separation convolution and standard convolution is shown in Figure 3.

Consider there is an input volume with width and height D_f, and the number of input channels M. If a color image

(a) (b)

Figure 1: Comparison of whether to use data smoothing in spicy crayfish images. (a) Original image. (b) Smoothed image.

Figure 2: Examples of images from the CF-108 dataset. Each row represents four images from a dish class. From top to bottom, the food names are scrambled eggs with pepper, noodles with sesame paste, and braised fish, respectively.

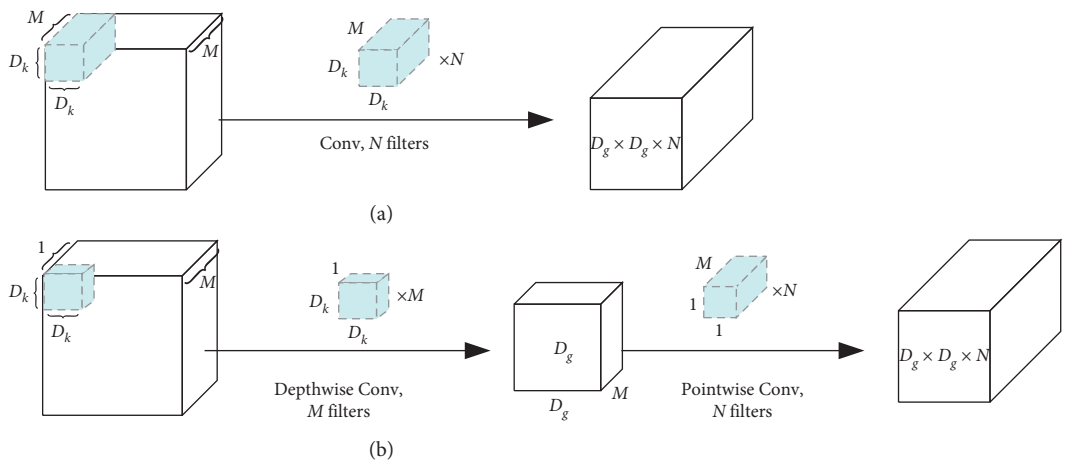


Figure 3: Comparison between standard convolution and depthwise separable convolution. (a) Standard convolution. (b) Depthwise separable convolution.

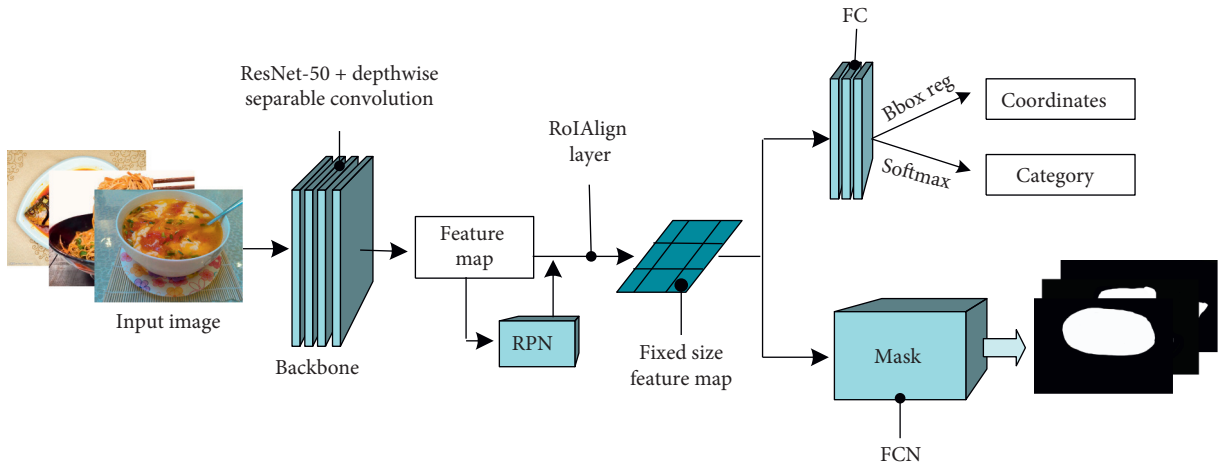


Figure 4: The training procedure of Mask R-DSCNN for the Chinese food image detection.

was an input, then M would be equal to three for the RGB channels. In standard convolution, the application of filters across all input channels and the combination of these values are done in a single step. As for N convolution kernels of shape $D_k * D_k * M$ that are applied on the input in standard convolution neural network, the output volume would be $D_g * D_g * N$. The cost of this convolution operation would be $N * D_k^2 * D_g^2 * M$. Taking the same input volume for comparison, depthwise separable convolution breaks the convolution down into two parts—depthwise convolution and pointwise convolution. Depthwise convolution applies convolution to a single input channel at a time. Therefore, each convolution kernel of shape $D_k * D_k * 1$ is applied to a single input channel in depthwise convolution stage with M such convolution kernels required over the entire input volume. Stacking the M outputs from each of these M convolutions together, an output volume with shape of $D_g * D_g * M$ is taken. Ending depthwise convolution, it will be succeeded by pointwise convolution, which involves performing the linear combination of each layer. The filter is basically $1*1$ convolution operation over all M layers. Assuming N such filters, the output volume will thus have the same shape as the standard convolution $D_g * D_g * N$. The total cost of these two phases would be $M * D_k^2 * D_g^2 + M * D_g^2 * N$, that is, $(D_k^2 + N) * M * D_g^2$. The effect of depthwise separable convolution can be shown as follows:

$$\frac{(D_k^2 + N) * M * D_g^2}{D_k^2 * M * N * D_g^2} = \frac{1}{N} + \frac{1}{D_k^2}. \quad (5)$$

For instance, considering the output feature volume N of 1024 and a kernel of size 3, the ratio is 0.112. In other words, standard convolution is nine times more than the number of multiplications. Therefore, we conclude that the computational resources required for depthwise separable convolution are much lower than the standard convolution.

4.2. Training Infrastructure. Mask R-CNN has high detection accuracy in image recognition and segmentation, but suffers from excessive computing resources and storage space. In this section, we use depthwise separable convolution instead of traditional convolution to reduce model consumption. Specifically, we replaced all convolutional blocks of ResNet-50 with depthwise separation convolution to complete feature extraction. Figure 4 illustrates the training procedure of Mask R-DSCNN for Chinese food image detection.

Table 2: Comparison of APs values on COCO dataset.

Models	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Mask R-CNN	0.381	0.553	0.426	0.185	0.407	0.561
Mask R-DSCNN	0.219	0.482	0.237	0.098	0.284	0.372

The training procedure of Mask R-DSCNN consists of three modules. The backbone is typically built by a depthwise separable convolution network with FPN architecture for feature maps extraction from input images. The feature map is shared for subsequent RPN layers and the RoIAlign layer. The RPN network is used to generate region proposals. This layer uses softmax to determine whether anchors are positive or negative and then uses bounding box regression to modify anchors to obtain accurate proposals. RoIAlign layer collects the input feature map and proposals, extracts the proposal feature maps after synthesizing the information, and then sends them to the subsequent multibranch prediction network. The FC layers use proposal feature maps to calculate the category of the proposal and use bounding box regression to obtain the final precise position of the detection box and the FCN segmented instance for masking.

5. Experiments

In this section, we conducted experiments for Chinese food image detection based on Mask R-DSCNN. First, the evaluation metrics and experimental settings are described, and then we assess the effectiveness of Mask R-DSCNN on the COCO dataset with Mask-RCNN for comparison. Finally, we provide the experimental results and analysis of Chinese food image detection.

Table 3: Model size and training speed comparison on COCO dataset.

Models	Model size (M)	Total running time (s)	Average computing time (s)
Mask R-CNN	245	5131	1.02
Mask R-DSCNN	93	3716	0.74

Table 4: Comparison of APs values on CF-108 dataset.

Models	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Mask R-CNN	0.576	0.881	0.648	0.279	0.682	0.875
Mask R-DSCNN	0.449	0.792	0.527	0.168	0.564	0.782

5.1. Implementation Details. To evaluate the performance of Mask R-DSCNN, we conduct a comparison of Mask R-DSCNN with the standard Mask R-CNN on the COCO dataset. The evaluation metrics of detection refer to the COCO target detection and evaluation indicators Average Precision (AP), which can effectively detect the similarity between the real target and the predicted target. As for the consumption of the model, it mainly refers to the model size and the time consumption for training. The experiments were conducted on NVIDIA Tesla M60 with Tensorflow2.0 [41] and Python 3.6.

5.2. The Effectiveness of Mask R-DSCNN. We trained Mask R-DSCNN on the COCO dataset and standard Mask R-CNN for comparison. The anchor size is set to (128, 256, 512), and the aspect ratio is set to (0.5, 1, 2). Stochastic gradient descent (SGD) is selected as the training optimizer. The learning rate was set at 0.001, momentum to 0.9, batch size to 128, and a total of 200,000 epochs. After training, we randomly selected 5,000 images from the testing set of the COCO dataset to evaluate the model performance (see Table 2).

Experiments show that the values of APs on Mask R-DSCNN are slightly lower but still on par with Mask R-CNN. This is because the Mask R-DSCNN replaces the standard convolutional layer with a deep separation convolution to extract features, which may cause some feature information loss. The model size and running time are recorded on the same configuration (see Table 3).

It can be seen that Mask R-DSCNN is more cost-efficient with smaller model size and thus benefits running speed. Explicitly, the model size of Mask R-CNN is 245 M, while the model size of Mask R-DSCNN is only 93 M, which is much lower than that of Mask R-CNN. In addition, the total running time of Mask R-DSCNN is 3716 s, which is more than 1400 s shorter than the running time of Mask R-CNN. Therefore, we can conclude that the Mask R-DSCNN can significantly reduce resource consumption and improve the detection efficiency without hurting too much accuracy.

5.3. Chinese Food Image Detection. To save training resources and ensure the model's performance on the Chinese food image, we fine-tuned both the Mask R-CNN and Mask R-DSCNN that are pretrained with the COCO dataset. Fine-tuning is a method to apply previously learned knowledge to new knowledge. In terms of deep learning, this means that the weight of each layer of the node is no longer randomly initialized, but is initialized using the trained model parameter layer. Since the target features extracted by deep learning are hierarchical, the high-level network will extract random combinations of features extracted by the low-level network. Therefore, the primary information extracted by deep learning is common in different datasets. If the training results obtained by the model on a large dataset are right, then the primary features obtained by the model can also be used on another dataset. In this experiment, the parameters of the models successfully trained on the COCO dataset are used as initialization parameters, and then we fine-tune the models with the new Chinese food image dataset. The size of the anchor is set to (8, 16, 32, 64, 128), and the aspect ratio is set to (0.5, 1, 2). The optimizer is SGD with a learning rate of 0.001 and a momentum of 0.9. The batch size is 64 with a total of 200,000 epochs. After training, we also randomly selected 5,000 images from the testing set of CF-108 dataset to evaluate the model performance (see Table 4).

Same as trained on COCO dataset, the Mask R-DSCNN model in CF-108 dataset training is slightly weaker but tolerable, and when the threshold of IOU is set at 0.5, the AP value of the two models is closest. Figure 5 shows the detection of Chinese food images by the Mask R-CNN model and the Mask R-DSCNN model when the IOU threshold is fixed at 0.5.

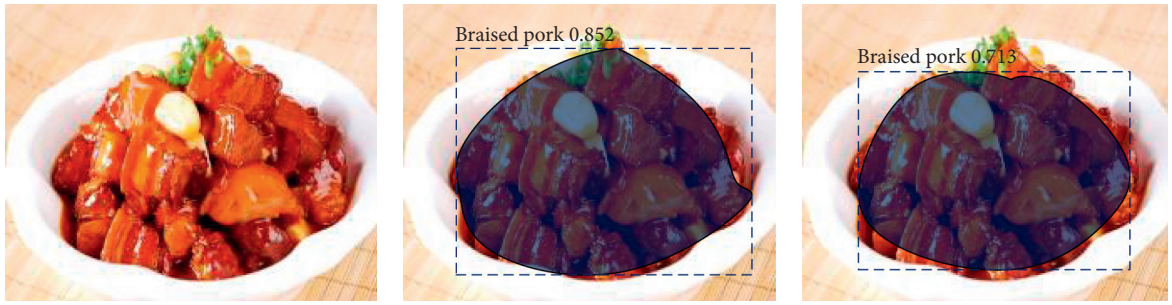
The results show that Mask R-DSCNN can successfully identify and segment braised pork, and the two models are almost the same in terms of regression box and mask survival rate.

The comparison on model size and running time between the two models is shown in Table 5.

It can be seen that Mask R-DSCNN still leads to a competitive result with smaller model size and shorter running time. Therefore, we can claim that Mask R-DSCNN has practical significance for Chinese food image detection.

6. Conclusions

In this paper, a method for Chinese food image detection based on an improved structure of Mask R-CNN was proposed. To achieve that goal, we first built a dataset of Chinese food images, called CF-108, which contains 100,800 images of 108 categories covering most Chinese food. In addition, we proposed a new model framework, namely, Mask R-DSCNN, with deep separable convolution instead of traditional convolution for reducing model consumption. The experiment results on the CF-108 dataset demonstrate that Mask R-DSCNN can greatly reduce the resource consumption and improve the detection efficiency of the Chinese food images without hurting too much accuracy.



(a) (b) (c)

Figure 5: The experimental results of Chinese food image detection. (a) Original image. (b) Prediction of Mask R-CNN. (c) Prediction of Mask R-DSCNN.

Table 5: Model size and training speed comparison on CF-108 dataset.

Models	Model size (M)	Total running time (s)	Average computing time (s)
Mask R-CNN	330	4517	0.90
Mask R-DSCNN	125	3244	0.64

Further work will be carried out with multispectral or hyperspectral images as in [42].

Data Availability

The dataset and software code used to support this study's findings have not been made available because the data also form part of an ongoing study. Requests for data, after publication of the ongoing study, will be considered by the corresponding author.

Disclosure

The authors contributed equally to this work and should be considered co-first authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (grant no. F060609).

References

[1] A. E. Mesas, M. Muñoz-Pareja, E. López-García, and F. Rodríguez-Artalejo, "Selected eating behaviours and excess body weight: a systematic review," *Obesity Reviews*, vol. 13, no. 2, pp. 106–135, 2012.

[2] M. B. E. Livingstone and A. E. Black, "Markers of the validity of reported energy intake," *The Journal of Nutrition*, vol. 133, no. 3, pp. 895S–920S, 2003.

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

[4] S. Mezgec and B. Koroušić Seljak, "NutriNet: a deep learning food and drink image recognition system for dietary assessment," *Nutrients*, vol. 9, no. 7, p. 657, 2017.

[5] K. Takahashi, K. Doman, Y. Kawanishi et al., "Estimation of the attractiveness of food photography focusing on main ingredients," in *Proceedings of the 9th Workshop on Multimedia for Cooking and Eating Activities in Conjunction with the 2017 International Joint Conference on Artificial Intelligence*, pp. 1–6, Melbourne, Australia, August 2017.

[6] C. Liu, Y. Cao, Y. Luo, G. Chen, V. Vokkarane, and Y. Ma, "Deepfood: deep learning-based food image recognition for computer-aided dietary assessment," in *Proceedings of the International Conference on Smart Homes and Health Telematics*, pp. 37–48, Wuhan, China, May 2016.

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

[8] R. Xu, L. Herranz, S. Jiang, S. Wang, X. Song, and R. Jain, "Geolocalized modeling for dish recognition," *IEEE Transactions on Multimedia*, vol. 17, no. 8, pp. 1187–1199, 2015.

[9] G. M. Farinella, D. Allegra, M. Moltisanti, F. Stanco, and S. Battiato, "Retrieval and classification of food images," *Computers in Biology and Medicine*, vol. 77, pp. 23–39, 2016.

[10] P. Yi, Z. Wang, K. Jiang, Z. Shao, and J. Ma, "Multi-temporal ultra dense memory network for video super-resolution," *IEEE Transactions on Circuits and Systems for Video Technology*, .

[11] J. Ma, W. Yu, P. Liang, C. Li, and J. Jiang, "FusionGAN: a generative adversarial network for infrared and visible image fusion," *Information Fusion*, vol. 48, pp. 11–26, 2019.

[12] J. Ma, X. Wang, and J. Jiang, "Image superresolution via dense discriminative network," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 7, pp. 5687–5695, 2020.

[13] H. Kagaya, K. Aizawa, and M. Ogawa, "Food detection and recognition using convolutional neural network," in *Proceedings of the 22nd ACM International Conference on Multimedia-MM'14*, pp. 1085–1088, Mountain View, CA, USA, June 2014.

[14] H. Hassannejad, G. Matrella, P. Ciampolini, I. De Munari, M. Mordonini, and S. Cagnoni, "Food image recognition using very deep convolutional networks," in *Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management-MADiMa '16*, pp. 41–49, Amsterdam, Netherlands, October 2016.

[15] A. Singla, L. Yuan, and T. Ebrahimi, "Food/non-food image classification and food categorization using pre-trained googlenet model," in *Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management-MADiMa '16*, pp. 3–11, Amsterdam, Netherlands, October 2016.

[16] X. Wu, D. Hong, J. Tian, J. Chanussot, W. Li, and R. Tao, "ORSIm detector: a novel object detection framework in optical remote sensing imagery using spatial-frequency channel features," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 7, pp. 5146–5158, 2019.

[17] X. Wu, D. Hong, J. Chanussot, Y. Xu, R. Tao, and Y. Wang, "Fourier-based rotation-invariant feature boosting: an efficient framework for geospatial object detection," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 2, pp. 302–306, 2019.

[18] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, Columbus, OH, USA, June 2014.

[19] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440–1448, Santiago, Chile, December 2015.

[20] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 91–99, Montreal, Canada, December 2015.

[21] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961–2969, Venice, Italy, October 2017.

[22] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, Las Vegas, NV, USA, June 2016.

[23] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7263–7271, Honolulu, HI, USA, July 2017.

[24] J. Redmon and A. Farhadi, "Yolov3: an incremental improvement," 2018, http://arxiv.org/abs/1804.02767.

[25] W. Liu, D. Anguelov, D. Erhan et al., "Ssd: single shot multibox detector," in *Computer Vision—ECCV 2016*, pp. 21–37, Springer, Cham, Switzerland, 2016.

[26] T.-Y. Lin, M. Maire, S. Belongie et al., "Microsoft coco: common objects in context," in *Computer Vision—ECCV 2014*, pp. 740–755, Springer, Cham, Switzerland, 2014.

[27] F. Chollet, "Xception: deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition CVPR)*, pp. 1251–1258, Honolulu, HI, USA, July 2017.

[28] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125, Honolulu, HI, USA, July 2017.

[29] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, Boston, MA, USA, June 2015.

[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 1097–1105, Lake Tahoe, NV, USA, December 2012.

[31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, http://arxiv.org/abs/1409.1556.

[32] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9, Boston, MA, USA, June 2015.

[33] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826, Las Vegas, NV, USA, June 2016.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, Las Vegas, NV, USA, June 2016.

[35] R. Girshick, F. Iandola, T. Darrell, and J. Malik, "Deformable part models are convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 437–446, Boston, MA, USA, June 2015.

[36] F. Menczer, G. Pant, and P. Srinivasan, "Topical web crawlers: evaluating adaptive algorithms," *ACM Transactions on Internet Technology*, vol. 4, no. 4, pp. 378–419, 2004.

[37] E. Rahm and H. H. Do, "Data cleaning: problems and current approaches," *IEEE Bulletin of the Technical. Committee on Data Engineering*, vol. 23, no. 4, pp. 3–13, 2000.

[38] B. I. Justusson, "Median filtering: statistical properties," *Two-Dimensional Digital Signal Prcessing II*, pp. 161–196, Springer, Berlin, Germany, 1981.

[39] X. Chen, Y. Zhu, H. Zhou, L. Diao, and D. Wang, "Chinesefoodnet: a large-scale image dataset for Chinese food recognition," 2017, http://arxiv.org/abs/1705.02743.

[40] A. G. Howard, M. Zhu, B. Chen et al., "Mobilenets: efficient convolutional neural networks for mobile vision applications," 2017, http://arxiv.org/abs/1704.04861.

[41] M. Abadi, P. Barham, J. Chen et al., "Tensorflow: a system for large-scale machine learning," in *Proceedings of the 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pp. 265–283, Savannah, GA, USA, September 2016.

[42] D. Hong, N. Yokoya, J. Chanussot, and X. X. Zhu, "An augmented linear mixing model to address spectral variability for hyperspectral unmixing," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 1923–1938, 2018.