ISSN: 2414-1895 DOI: 10.6919/ICJE.202111_7(11).0074

Chinese Common Food Classification by Convolution Neural Network

Yusong Wang[1 *], Chang Wei [2, a], Sice Huang[3, b], Yuning Zhou [4, c]

[1] Guangxi University of Science and Technology, Liuzhou, Guangxi, China

[2] The Hong Kong Polytechnic University, Hongkong, China

[3] University of Electronic Science and Technology of China, Chengdu, Sichuan, China

[4] Jiangsu University, Zhenjiang, Jiangsu, China

[*]Corresponding author e-mail:183363200@qq.com, [a] 245387176@qq.com,

[b] 826225118@qq.com, [c] 530906899@qq.com

These authors contributed equally to this work

Abstract

The article reviews the progress and application of food/dish recognition through Convolution Neural Network (CNN) technique. Also, the article explores the CNN architecture in an experiment to classify four types of common Chinese dishes and reaches an accuracy of 90% after tuning hyperparameters and applying different strategies of optimization.

Keywords

Progress and application; Convolution Neural Network (CNN); Chinese dishes; optimization.

1. Introduction

The type of Chinese dishes is tremendously various, and the color, texture and appearance are diverse due to the different cooking methods. This is to say that the identification of Chinese dishes is difficult, even for human beings to recognize when they are having orders in school cafeteria. The aim of the report is to research extraction and identification of dishes in order to save time for students. This report will apply deep learning technique of Convolutional Neural Network (CNN) to achieve this aim with relevant packages like tensorflow. It refers to data set making and identification of four typical Chinese dishes as examples. This report will focus on the method of data set making and CNN model building and optimization. Through constructing and optimizing CNN model, the accuracy of identification is satisfyingly over 90%.

2. Background

The burgeoning of deep learning technology has revolutionized the area of object recognition by computer, based on images or videos. The more accurate and fast recognition realized with Convolutional Neural Network (CNN), a deep learning tool, prompts the advance especially in computer vision and artificial intelligence since the image recognition and classification is a vital mean of input for further tasks done by the computers [1].

Among various objects, food recognition has caused intensive interests due to its wide applications. One of the major concerns in this field is dietary assessment: people's food intake will be organized for nutritional plans both inside and outside a clinical environment [2]. By virtue of CNN architecture,

ISSN: 2414-1895 DOI: 10.6919/ICJE.202111_7(11).0074

the traditional methods done by recalling and logging can be replaced by more efficient and accurate food recognition via mobile camera, which does not have problems of memory and human recording [3]. Another popular application is in the canteens and food courts where each dish is charged based on its category [4]. And it is conceivable that an automated charging system based on dish recognition will reduce the chance of mistakes and save lots of human labor and time. Aside from those, the technology of food recognition can be further coupled into the design of AI culinary devices [5], including products like AI smart oven and AI cooking robot, as the necessary means of data input.

Despite the widespread potential applications, food or dish as a special object pose challenges different from other common objects like cars, animals, and other rigid objects: the specialty is that dish classes have a much higher inter-class similarity and intra-class variation [6]. Considering different kinds of leaf vegetables, some of them might not have a distinct difference with each other so that many people might confuse with them even before they are cooked into dishes. In addition, a similar cooking recipe on different raw materials also tend to homogenize the dishes' appearance, causing extra difficulty in image classification. From another aspects, considering salad and tofu, those are a mixture of different ingredients and also in different shapes, sizes, and probably colors, depending on regional culinary customs. However, the features of a particular food or dish could be readily recognizable by human eyes even in a single image, regardless of the significance variations in appearance, given that this person has tried on some, but definitely not all, of the food in several different versions [7]. Therefore, despite the more complex pictorial variation, there are still plenty of information and features behind to be dug out for the task of classification.

The research of using deep learning in food recognition is relatively new and it begins in 2014. The most commonly applied deep learning architecture is deep CNN, and its advantage compared with conventional method is evident. In 2014, one of the first works that employed CNN achieved a classification on foods with the accuracy of 72.26% using a model similar to AlexNet [8]. But same in 2014, Bossard et all. Achieved only 50.76% classification accuracy with the use of random forest technique on the Food101 dataset, and they also conclude that their achieved accuracy cannot outperform CNN [9]. From then on, more research teams work on this field, developing their own food dataset, optimizing the CNN model, and pushing the classification accuracy higher into more than 80% [10, 11]. However, such an accuracy is still not sufficient to generate reliable applications in related fields.

3. Research

3.1 Computation Environment

The hardware configuration of the computer is tabulated as follow:

Table I. Computation Environment

CPU	i7 10750H
GPU	RTX 2070s
Memory	32G
OS	Windows 10
Deep Learning Framework	tensorflow.keras
CUDA	10.1

3.2 Data Collection:

Data collection is important to data set making because the amount and quality of pictures collected will significantly influence the quality of data set, which further determines the fitting in training regardless of the CNN architecture [12]. In this part, a web crawler is constructed to collect the images from Internet by using packages 're' and 'BeautifulSoup'. The image source crawled is BaiduImage. In this stage, 1,000 pictures of four typical Chinese dishes (rice, shredded potatoes, steamed bass and Mapo Tofu) are collected and 820 qualified pictures of each are saved after manual selection to ensure the resolution of the image source.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202111_7(11).0074

3.3 Data Processing

All images are digitized to tensor and cropped to the size of 32*32 by using 'opencv' package in python, and this is for feeding into the CNN model and to reduce the amount of computation in the model. Also, each image is labeled according to its category of four dishes for the sake of training set and testing set.

Finally, all images with their labels are packaged as a 'mat' file by calling 'savemat' method in the package 'scipy.io'. The dataset in '.mat' format is essentially a python dictionary. The 'X' key saves the tensors which represent images and the 'y' key saves the labels of each image. The class of steamed bass, rice, Mapo Tofu and shredded potatoes are labeled as integer '1', '2', '3' and '4', respectively. The ratio of the training set to the test set is about four to one and the number of pictures in train set and test set are 649*4 and 173*4 separately.

Fig. 1. Original image crawled from Internet

Fig. 2. Processed images

3.4 Design and Construction of CNN Model

The structure of this model is constituted by convolutional layers, activation function, pooling layer, and fully connected layer. Convolutional layer is used to extract the features of the image. Activation function is used to add nonlinearity, simulating what the neurons do in physiology. In this part, relu function is used for simplicity in calculation. Pooling layer is used to reduce the dimension of images but retains important information. Fully connected layer does the final classification based on the extracted features [13].

Specifically, the CNN architecture used in this work is INPUT -> CONV1 -> CONV2 -> CONV3 -> POOL1 -> CONV4 -> FC1 -> FC2->OUTPUT. There are four convolution layers with activation function 'relu', one max pooling layer between the third and forth convolution layers, two fully connected layers where the first layer uses relu and the second uses softmax for classification

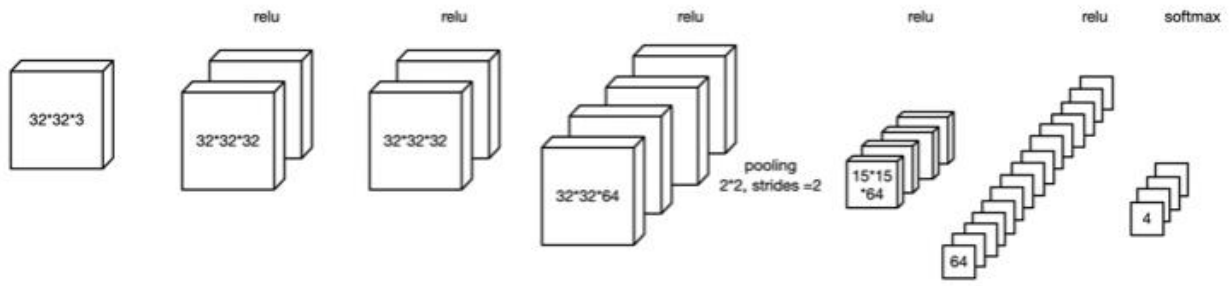

Fig. 3. CNN model demonstration

The specifications of different layers are as follow:

- CONV1: The depth of filters is 32, kernel size is 3*3, activation function is relu.
- CONV2: The depth of filters is 32, kernel size is 3*3, activation function is relu.
- CONV3: The depth of filters is 64, kernel size is 3*3, activation function is relu.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202111_7(11).0074

- CONV4: The depth of filters is 64, kernel size is 3*3, activation function is relu.
- POOL1: pooling ratio is 2*2.
- FC1: The number of neurons is 64, activation function is relu.
- FC2: The number of neurons is 4, activation function is softmax.

3.5 Programming

All programming is done in python. The versions of python and tensorflow are 3.6 and 2.3.1, respectively. Two separate python files are created: one for loading data and data preprocessing, the other for CNN model buildup and feeding the data.

3.6 E1. Loading and preprocessing

In the file of data loading, the packages imported are 'numpy' and 'scipy.io' separately. The first step is to load the mat file of training set and testing set by using 'load' function from 'scipy.io'. Note that the first element of the dataset which is from dataset generation should be deleted, since it is irrelevant and interfering the classification.

The shape of each set can be seen clearly. The shape of training set is (2596, 32, 32, 3) -- number of pictures, width of pictures, high of pictures, the channel of pictures. And the shape of the training label is (2596, 1) -- the number of corresponding labels, the size of each label. The shape of testing set is (692, 32, 32, 3) -- number of pictures, width of pictures, high of pictures, the channel of pictures. The shape of testing label is (692, 1) -- the number of labels, the size of each label.

The next step is to do normalization for the pictures of these set. This maps the elements of matrices representing the pictures to the range of [-1,1], reducing the color variability for better generalization and making data computation more convenient and faster [14]. The processed dataset will be fed into the CNN model in the second python file.

3.7 E2. CNN Modeling

In building the architecture, functions, including Conv2D, MaxPooling2D, Flatten, Dense and Dropout, the L2 regularization as an argument, from 'tensorflow.keras.layers', 'numpy' are called, and the processed data are imported.

After the final FC2 layer, the output from the model is compared with the preset label in the range of 0 to X-1. Therefore, the number of nodes is set to 5 rather than 4.

In the model, dropout strategy is used to avoid overfitting by setting the number of some nodes to 0 and the dropout rate is set to 0.5 [15]. Also, kernel regularization (L2 regularization) is used as well. Flatten makes multidimensional input one-dimensional, which is often used in the transition from convolution layer to fully connected layer. Flatten layer does not affect the size of the batch.

The second step is to compile the model created before, the optimizer and loss functions are defined. The learning rate is set to 0.001 after tuning, and 'sparse_categorical_crossentropy' is used for loss function to match the format of our one-integer label.

The next stage is to build function for training. The train set will be sliced randomly. The total epoch of training of sliced set is set to 5 and the batch size is 200 (pictures). 10% pictures of sliced set will be used for validation. The total round for training is 80.

In the final stage, the trained model will be tested by using testing set and calling 'evaluate' function.

```
Epoch 2/5
 1/12 [=>............................] - ETA: 0s - loss: 0.0891 - acc: 0.9900
 2/12 [====>.........................] - ETA: 1s - loss: 0.1078 - acc: 0.9850
 3/12 [======>.......................] - ETA: 1s - loss: 0.1045 - acc: 0.9883
 4/12 [=========>....................] - ETA: 1s - loss: 0.1069 - acc: 0.9875
 5/12 [===========>..................] - ETA: 1s - loss: 0.1015 - acc: 0.9900
 6/12 [==============>...............] - ETA: 1s - loss: 0.1030 - acc: 0.9892
 7/12 [================>.............] - ETA: 1s - loss: 0.1096 - acc: 0.9857
 8/12 [==================>...........] - ETA: 0s - loss: 0.1075 - acc: 0.9869
 9/12 [====================>.........] - ETA: 0s - loss: 0.1048 - acc: 0.9872
10/12 [======================>.....] - ETA: 0s - loss: 0.1075 - acc: 0.9860
11/12 [========================>...] - ETA: 0s - loss: 0.1101 - acc: 0.9850
12/12 [==============================] - ETA: 0s - loss: 0.1087 - acc: 0.9854
12/12 [==============================] - 3s 237ms/step - loss: 0.1087 - acc: 0.9854 - val_loss: 0.1922 - val_acc: 0.9462
```

Fig. 4. Example of training and testing

ISSN: 2414-1895 DOI: 10.6919/ICJE.202111_7(11).0074

4. Conclusion

CNN can simplify complex problems and reduce a large number of parameters into a small number of parameters, which is beneficial to process and retain image features. In this research, the details and characteristics of dish image are analyzed and then a new CNN model is designed based on the structure of the traditional convolution neural network. This CNN model can extract and retain the image features well and deal with the problem of over fitting to some extent through configuring the number of convolution layers, pooling layers and nodes reasonably. After a few rounds of training, the accuracy of identification is satisfying and the average accuracy is over 90%. In this process, the over fitting is addressed.

In order to verify the validity and rationality of CNN model referred in this report, and achieve the goal of identifying Chinese dishes, a picture set containing four different types of Chinese dishes is built. All pictures are collected by crawler set up by python script and the unrelated pictures are filtered by manual selection, which guarantees the quality of training and testing. The effectiveness of the proposed CNN model is proved by experimenting on this CNN model and dataset.

There are many types of Chinese food and eight major regional Chinese food covering thousands of dishes, which always influences the dining efficiency of students negatively when they eat in the school canteen. This report designs and constructs a new CNN model to identify Chinese dishes, which can reduce the unnecessary time on food identification after applying. The future research should be on identification of pictures with high similarity, aiming for similar in appearance but different dishes. Also, the accuracy could be further improved. Food identification has great prospects with the development of artificial intelligence. Under this situation, how to apply it in the real life effectively is the main research directions.

References

[1] Krizhevsky, A.; Sutskever, I. & Hinton, G. E. (2012), ImageNet Classification with Deep Convolutional Neural Networks, in F. Pereira; C. J. C. Burges; L. Bottou & K. Q. Weinberger, ed., 'Advances in Neural Information Processing Systems 25', Curran Associates, Inc., pp. 1097--1105.

[2] Mezgec, S. and Koroušić Seljak, B. (2017) 'NutriNet: A Deep Learning Food and Drink Image Recognition System for Dietary Assessment', Nutrients, 9(7). doi: 10.3390/nu9070657.

[3] Katz, D. L. et al. (2020) 'Dietary assessment can be based on pattern recognition rather than recall', Medical Hypotheses, 140. doi: 10.1016/j.mehy.2020.109644.

[4] Wang, Y. et al. (2019) 'Mixed dish recognition through multi-label learning', CEA 2019 - Proceedings of the 11th Workshop on Multimedia for Cooking and Eating Activities, pp. 1–8. doi: 10.1145/3326458.3326929.

[5] PR Newswire (2018) 'Midea launches One of the Largest Chinese Dish AI Datasets and High Precision AI Deep Learning Models for Chinese Dish Recognition'.

[6] Aguilar, E., Bolaños, M. and Radeva, P. (2017) 'Food Recognition using Fusion of Classifiers based on CNNs'. doi: 10.1007/978-3-319-68548-9_20.

[7] Kiourt, C., Pavlidis, G. and Markantonatou, S. (2020) 'Deep learning approaches in food recognition'.

[8] Kawano, Y., & Yanai, K. (2014b). Food Image Recognition with Deep Convolutional Features. Proc. of ACM UbiComp Workshop on Cooking and Eating Activities (CEA), (pp. 589-593).

[9] Bossard L., Guillaumin M., Van Gool L. (2014) Food-101 – Mining Discriminative Components with Random Forests. In: Fleet D., Pajdla T., Schiele B., Tuytelaars T. (eds) Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol 8694. Springer, Cham. https://doi.org/10.1007/978-3-319-10599-4_29.

[10] Singla, A., Yuan, L. and Ebrahimi, T. (2016) 'Food/non-food image classification and food categorization using pre-trained GoogLeNet model', MADiMa 2016 - Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management, co-located with ACM Multimedia 2016, pp. 3–11. doi: 10.1145/2986035.2986039.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202111_7(11).0074

[11] Park, S. J., Palvanov, A., Lee, C. H., Jeong, N., Cho, Y. I., & Lee, H. J. (2019). The development of food image detection and recognition model of Korean food for mobile dietary management. Nutrition research and practice, 13(6), 521–528.

[12] Najafabadi, M.M., Villanustre, F., Khoshgoftaar, T.M. et al. (2015) Deep learning applications and challenges in big data analytics. Journal of Big Data 2, 1 (2015). https://doi.org/10.1186/s40537-014-0007-7.

[13] Yamashita, R. et al. (2018) 'Convolutional neural networks: an overview and application in radiology', Insights into Imaging, 9(4), p. 611. doi: 10.1007/s13244-018-0639-9.

[14] Justin Tyler Pontalba et al. (2019) 'Assessing the Impact of Color Normalization in Convolutional Neural Network-Based Nuclei Segmentation Frameworks', Frontiers in Bioengineering and Biotechnology, 7. doi: 10.3389/fbioe.2019.00300.

[15] Srivastava, N. et al. (2014) 'Dropout: a simple way to prevent neural networks from overfitting', Journal of Machine Learning Research (JMLR), 15, p. 1929.