Advances in Engineering Research (AER), volume 124
2nd International Symposium on Advances in Electrical, Electronics and Computer Engineering
(ISAEECE 2017)

Automatic Cashier System Based on Meal Plate Detection Using Deep Learning

Weibin Zheng[1, a], Shengliang Peng[1, b], Qilong Chen[1, c], Hao Ouyang[1, d], Shiheng Lin[1, e]

Institute of Communications Technology, Huaqiao University, Xiamen 361021, China[1]

[a]wb_zheng@outlook.com, [b]peng.shengliang@gmail.com, [c]936964961@qq.com，

[d]oyang9595@outlook.com, [e]hp677@qq.com

Keywords: Automatic Cashier System, Meal Plate Detection, Deep Learning, Faster-R-CNN

Abstract. Deep learning (DL) is an important branch of machine learning (ML) that has excellent performance in recognition and detection tasks. Currently, most self-service restaurants/canteens are utilizing traditional manual cashier approaches, which suffer from the problems of error calculation, low efficiency and high labor cost. In this paper, an automatic cashier system using DL is designed to solve the problems mentioned above. This system recognizes the numbers and types of multiple plates in a meal tray with faster region based convolutional neural network (Faster-R-CNN) and automatically calculates the total price of the meal. Experimental results show the satisfying meal plate detection accuracy and high cashier speed of our system.

Introduction

Deep learning (DL) is an important branch of machine learning (ML) that has a state-of-the-art ability for object detection. It has been widely used in many fields, such as artificial intelligence (AI), image recognition, speech recognition etc. in recent years [1]. The deep learning idea was introduced in literature [2] to solve the bottleneck of traditional neural networks. There are two main contributions in this literature: it shows that deep artificial neural network has the capability of autonomous feature learning and the learnt features are more representative than traditional hand-crafted features; it puts forward a step-by-step solution to conquer the convergence problem in the training stage. In 2012 ImageNet large scale visual recognition challenge (ILSVRC2012)**,** an eight-layer deep convolutional neural network (CNN) named AlexNet [3] was designed, which made a ten percent reduction of the top-5 error rate and created the best results in the history of this recognition game.

On the other hand, self-service restaurants are very popular in our daily lives. This type of restaurants prefills different meal plates with different kinds of food, and charges customers by identifying the categories and numbers of the meal plates they choose. This work is usually done by a cashier and has three major problems. First, the speed of checkout is slow and customers have to wait for a long time at peak dining period. Second, cashiers should remember the prices of all kinds of food, count the quantity of meal plates, and calculate the total price accurately, in which there inevitably exist some mistakes that cause great troubles to both cashiers and customers. Third, restaurants need to hire cashiers to handle this work, increasing the human resource cost. Automatic cashier system is designed to combat these problems. In literature [8], a radio frequency identification (RFID) based method was proposed. However, in this method, a RFID chip should be installed in each plate, resulting in additional cost and maintenance difficulty to the meal plates.

A good solution to the automatic cashier task is the use of computer vision based object detection. Generally, the traditional object detection is divided into three stages: region selection, feature extraction and classification. Region selection aims to determine the position of the object to be detected in the image. It uses a small sliding window to traverse the whole image in a fixed step length and finds out all possible positions of the object. Its disadvantages are also obvious: spending too much computing time and seriously affecting the detection speed. Feature extraction is the key stage of object detection. Conventional hand-crafted feature extraction methods, such as histogram of oriented gradients (HOG) [4] and local binary pattern (LBP) [5], usually require strong expertise and

a large amount of data to choose proper features, and are not very robust for obscure object detection as well as in complex scenarios. In classification stage, various classifiers can be utilized, including cascade classifier [6] and supported vector machine (SVM) [7].

This paper focuses on object detection using deep learning and designs an automatic cashier system for self-service restaurants. Massive images of meal plates were collected and labeled in advance. Then we feed these images to a CNN for training. The trained CNN model is transplanted to an embedded system equipped with a camera. In application, the embedded system captures an image of meal plates, recognizes plates with the trained model, and calculates the total price based on the preset unit price of each type of plates. This system does not require special meal plates equipped with RFID chips, and is superior in equipment cost and checkout speed.

The rest of this paper is organized as follows. Section II introduces the prevalent region based DL networks, including region based CNN (R-CNN), Fast-R-CNN and Faster-R-CNN. Section III describes the system design, network selection and embedded platform. Section IV illustrates the detailed steps of system realization. Test results are provided in Section V. Section VI concludes this paper.

Region-based Deep Learning Networks

R-CNN In 2014, literature [9] combined region proposal with CNN to replace traditional object detection approaches. It used selective search (SS) [10] method to traverse input images, extracted about 2000 region proposals to feed CNN. Feature extraction was carried out by network automatically. In addition, R-CNN designed a regression model to bounding box regression. This groundbreaking method could greatly improve the mean average precision (mAP) of pattern analysis, statistical modelling and computational learning, visual object classes (PASCAL VOC), but had a serious shortcoming: every region proposal needed feature extraction which consumed redundant computation and unnecessary computing time. In the experiments performed by authors, R-CNN took about 47 seconds to extract features, obviously far from the goal of real-time detection. In addition, R-CNN had to adjust region proposals to cater for CNNs. In that case, it would cause image distortion and affect the final results.

Fast-R-CNN In order to solve the problems given above, Fast-R-CNN [11] was proposed by adding region of interest (RoI) pooling layer on the basis of R-CNN structure. RoI layer mapped the different-size input to a fixed-scale feature vector. Region proposals were changed into a fixed-length feature vector. At the meantime, Fast-R-CNN combined the bounding box regression process with CNN, and shared convolution features to form a multi-task loss learning strategy. In addition, it added single scale testing and singular value decomposition (SVD) of fully connected layer in CNN to improve the detection speed. The experimental results in literature [11] showed that Fast-R-CNN not only improved the mAP of PASCAL VOC, but also increased the detection speed by 213 times and 10 times compared with R-CNN and spatial pyramid pooling network (SPPNet) [12], respectively.

Faster-R-CNN After solving the problem of excessive computation in the process of feature extraction, how to reduce the time of generating region proposals becomes a key problem. Faster-R-CNN [13] indicated that there existed some information which could be used to generate the region proposal in each layer of CNN, and designed a new region proposal network (RPN). Faster-R-CNN is divided into two parts: Fast-R-CNN part and RPN part. RPN performed bounding box regression task and calculated objectness [14] score. The high scored region proposals were chosen to feed Fast-R-CNN detection net. RPN shared convolution feature maps with CNN, which significantly reduced extra time cost in region proposal generation stage. A four-stage training strategy was also developed in Faster-R-CNN to unify RPN and Fast-R-CNN. Region proposals and feature extraction use the same convolution feature maps and converged quickly while minimizing the generation time of region proposals. Moreover, Faster-R-CNN further improved the detection accuracy and speed.

System Design

Two major tasks of our automatic cashier system are concerned with detection network and embedded platform, respectively. In the first task, we select a detection network suitable for meal plate detection and feed the chosen network with meal plate pictures for training. In the second task, we transplant the trained network to an embedded platform, with which the picture of meal plates can be captured, the types of plates are detected, and finally the total price is calculated.

Network Selection

We choose RPN + Zeiler & Fergus Net (ZFNet) [15] in our system. ZFNet is an eight-layer CNN (not including data input layer). It uses deconvolution network to visualize the features of convolution layer in AlexNet. Convolution kernel size and stride parameters in first two convolution layers of AlexNet were adjusted to retain more original pixel information. The structure of meal plate detection network in our system is shown in Figure 1.

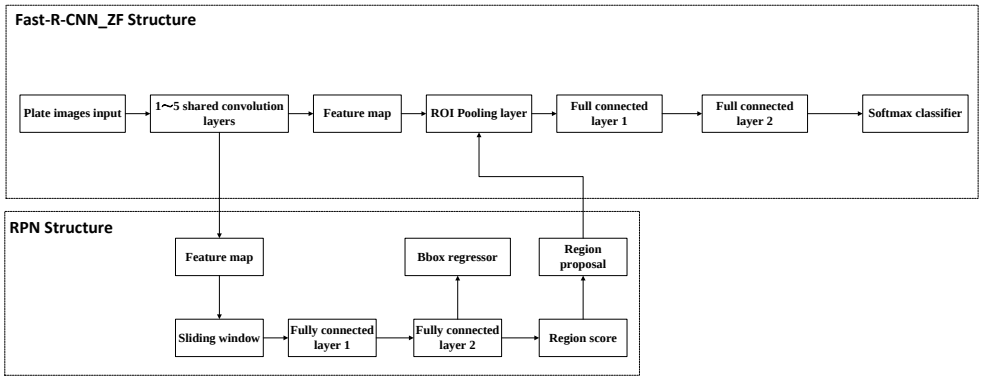

Fig. 1 Structure of meal plate detection network

From the chart, we can see that ZFNet and RPN share convolution feature maps of first five convolution layers. The feature map of fifth convolution layer is treated as the input of RPN. Scores of region proposals are obtained through one convolution layer and two fully connected layers. And we select top-300 high scored regions as region proposals to feed RoI pooling layer. Classifier determines maximum-probability category and region bounding box via several fully connected layers. There are 15 categories totally in meal plate detection network, including 14 meal plate categories (according to colors and the shapes of plates) and one background category.

Embedded Platform Selection

We choose JETSON TK1 mobile supercomputer. It was developed by NVIDIA and equipped with ARM Cortex-A15 CPU and NVIDIA Kepler GPU which contains 192 CUDA kernels. Furthermore, TK1 supports some common interfaces for developers, such as USB host, HDMI, PCLe Ethernet and so on. Figure 2 shows the interface diagram of JETSON TK1.

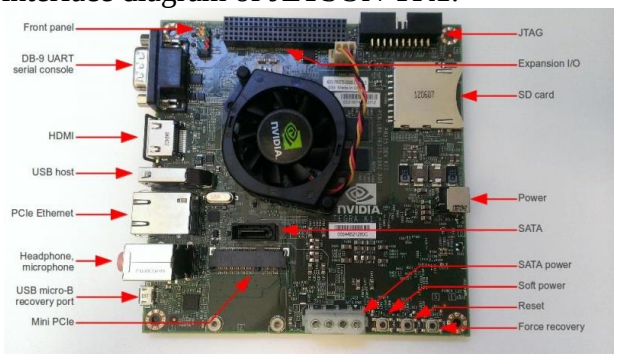

Fig. 2 Interface diagram of JETSON TK1

System Realization

Dataset Preparation In dataset preparation phase, we take more than 4200 meal plate pictures from a self-service restaurant through mobile phones. All these pictures are adjusted to the resolution of 640*480 pixels. Supervised learning is considered in our system, so the datasets need to be labeled before used for training. We choose LabelImg [16] annotation software to manually position objects for each image in the datasets and label the category. This software will generate extensible markup language (XML) files with the same format as PASCAL VOC sets after labeling. Image folder, filename and the upper-left and lower-right corner coordinates of the object region are stored in the XML file. Labeled images are divided into train sets and test sets according to the proportion of 8:2, while the image filenames are saved in a text document. Partial datasets shown in Figure 3.

Fig. 3 Partial dataset presentation

Training Process

The deep learning framework of is implemented based on Caffe [17] and py-faster-r-cnn in an Ubuntu system. Training process is divided into four stages: stage_1_rpn, stage_1_fast_rcnn, stage_2_rpn, stage_2_fast_rcnn. Maximum number of iterations for these stages is [50000 100000 50000 100000], respectively. Finally we get trained model named ZF_Final.caffemodel at the end of the fourth stage.

Test Results

In the test phase, the trained model is utilized by a classification algorithm written by Python running in the embedded platform. A camera is connected to the platform via USB, with which photos of meal plates can be captured and transmitted to the platform. Then the categories of meal plates is classified with the trained model and the total price is calculated accordingly. The entire checkout process takes about two seconds, which greatly improves the checkout speed. Figure 4 shows some detection results.

Fig. 4 Detection results

The feasibility of our automatic cashier system greatly depends on the performance of trained network. In order to verify validity of the meal plate detection network we selected about 850 meal plate pictures as the test set. All pictures in the test set are labeled previously and do not participate in the training process. Testing procedure is designed to traversing all pictures in test set and output the bounding box of each object and the category with maximum probability. Average precision (AP) of each category and mean average precision (mAP) are obtained and shown in table 1:

Table 1 Average precision for each category and mean average precision

Categories	Orange_cir	Yellow_squ	Pink_cir	Boat	White_cir	Iron_cir	Purple_cir
AP(%)	99.6	90.8	89.5	90.7	90.5	99.9	99.0
White_tri	Green_squ	Purple_squ	Green_ell	Yellow_cir	Blue_cir	White_ell	mAP(%)
98.6	89.7	90.9	98.1	96.6	99.8	100	95.5

As can be seen from table 1, the mAP of test sets can reach 95.5%, and average precision of each category is in high level. From the perspective of detection speed, it takes about 0.2 seconds to detect each meal plate image. In conclusion, our automatic cashier system is able to complete the checkout process accurately and quickly as expected.

Conclusions

We design an automatic cashier system for self-service restaurants based on DL in this paper. Faster-R-CNN detection network is exploited to automatically classify meal plate categories and the total price is calculated on an embedded platform. Comparing with manual cashier approaches, the new automatic cashier system proposed in this paper has made a great progress both in detection accuracy and checkout speed. The mAP of trained model and test results show the feasibility of this system. Future improvement is to promote detection accuracy by enlarging train sets and adjusting network structures. On the other hand, we will try to transplant this automatic cashier system to a mobile terminal, because completion of the checkout process by mobile phones could bring great convenience to daily life.

Acknowledgements

The authors would like to acknowledge the Jiangsu Province Postdoctoral Scientific Research Project (No. 1402041B), the Scientific Research Fund of Hunan Provincial Education Department (No. 16A174), and the Huaqiao University (No. 13BS101) for their financial supports.

References

[1] Geoffrey Hinton, LeCun Yann, Yoshua Bengio. Deep Learning [J]. Naure ,2015, 521, 436-444

[2] Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets[J]. Neural computation, 2006, 18(7): 1527-1554.

[3] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton ImageNet Classification with Deep Convolutional Neural Networks

[4]NavneetDalal, Bill Triggs. Histograms of Oriented Gradients for Human Detection[C]. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2005, 1: 886-893

[5] Yadong MU, Shuicheng Yan, Yi Liu, et al. Discriminative Local Binary Patterns for Human Detection in Personal Album[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008, 1-8.

[6] Paul Viola, Michael J. Jones. Robust Real-Time Face Detection[C]. IEEE International Conference on Computer Vision (ICCV),2001, 57(2): 137-154.

[7] Cortes, C. and Vapnik, V. (1995) Support-vector networks. Machine Learning, 20, 273-297.

[8] Mr.P. Chandrasekar, Mr.P. Chandrasekar. Smart Shopping Cart with Automatic Charging System

through RFID and ZigBee. ICICES2014 - S.A.Engineering College, Chennai, Tamil Nadu, India.

[9] Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. IEEE Conference on Computer Vision and Pattern Recognition, 2014.

[10] J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders. Selective search for object detection. IJCV, 2013.

[11] Ross Girshick. Fast R-CNN. IEEE International Conference on Computer Vision. 2015

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 9, pp. 1904–1916, 2015.

[13] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2016.

[14] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. PAMI, 2012.

[15] Zeiler M D, Fergus R. Visualizing and Understanding Convolutional Networks[J]. 2014, 8689:818-833.

[16] https://github.com/tzutalin/labelImg

[17] http://caffe.berkeleyvision.org/