Analysis of Commodity image recognition based on deep learning

Shuyan Wang1, 2
1. Faculty of Information Technology, Zhejiang University, Hangzhou 3100272. SWJTU-Leeds Joint School, Southwest Jiaotong University, Chengdu 611756

Li Xie*
Department of Instrument Science and Engineering, Zhejiang University, Hangzhou 310027

Lili Zhao
College of Control Science and Engineering, Zhejiang University, Hangzhou 310027

ABSTRACT

Deep learning has developed rapidly in recent years, especially in the field of image recognition. In this paper, the commodity recognition based on object detection method using deep convolutional neutral networks is investigated. Firstly, the commodity image dataset in real-world retail product checkout situations is constructed. Then, the image data is trained via object detection deep networks. Finally, three representative deep learning methods involving YOLOv3, Faster R-CNN and RetinaNet are analyzed in detail. The experimental results show the effectiveness of our proposed approach.

CCS CONCEPTS

• **Computing methodologies**; • **Artificial intelligence**; • **Computer vision**; • **Computer vision problems**; • **Object detection**;

KEYWORDS

Commodity recognition, object detection, deep convolutional neutral networks, product image synthesis

ACM Reference Format:
Shuyan Wang1, 2, Li Xie*, and Lili Zhao. 2021. Analysis of Commodity image recognition based on deep learning. In *2021 6th International Conference on Multimedia and Image Processing (ICMIP 2021), January 08–10, 2021, Zhuhai, China.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3449388.3449389

1 INTRODUCTION

Supermarkets and convenience stores provide much convenience to customers. However, the cost of retail stores is quite high resulted from high rents and labor costs. Therefore, finding measures to increase the efficiency of commodity recognition becomes crucial [1].

Two traditional methods of commodity recognition are barcode identification and RFID. Barcode identification is based on photoelectric conversion. Barcodes store encoded product information which can be decoded using code scanning devices. The accuracy of barcode identification is restricted due to the limited amount of information one barcode can store. Besides, the automation of barcode identification is rather low as the barcode sticking and scanning processes are completed manual. An alternate measure is RFID which makes uses of radio frequency signals to pass data and information automatically with no contact. RFID tags store the unique product ID and other commodity information. The advantages of RFID technology are that data reading and writing are convenient and RFID tags can store more information than barcodes. However, the costs of RFID tags are quite high.

As deep learning boosts at a staggering rate, object detection methods based on convolutional neutral networks are gradually applied to the area of commodity recognition [2]. They are much more automated than barcode identification. Object detection methods achieve commodity recognition by taking a picture of all products placed on the checkout platform and inputting the image into pretrained neutral networks, which extract features and perform object classification and bounding box regression. The whole detection process is completed by computers without human participation which saves plenty of labor forces. Besides, object detection algorithms can recognize multiple products on one input image at the same time, rather than scanning the code one by one for each product like the barcode identification method, which helps to increase the checkout frequency.

In this paper, the issue of commodity recognition is investigated based on deep object detection networks. The rest of the paper is organized as follows. In Section 2, we give a brief discussion on several typical deep learning networks on object detection. Then, the commodity image dataset in accordance with real-world recognition situations is construct in Section 3. We train the above object typical object detection networks on the dataset and compares their performance in Section 4. Finally, Section 5 concludes the paper with a brief summary.

2 COMMODITY RECOGNITION BASD ON DEEP NETWORKS

Different deep object detectors can be classified into one-stage detection framework and two-stage detection framework according to their network structures. Two-stage detectors are based on region proposals and realizes object classification through convolutional neutral network. While one-stage object detectors locate the boundaries of objects by regression. Common one-stage object detection

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ICMIP 2021, January 08–10, 2021, Zhuhai, China
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8916-7/21/01…$15.00
https://doi.org/10.1145/3449388.3449389

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

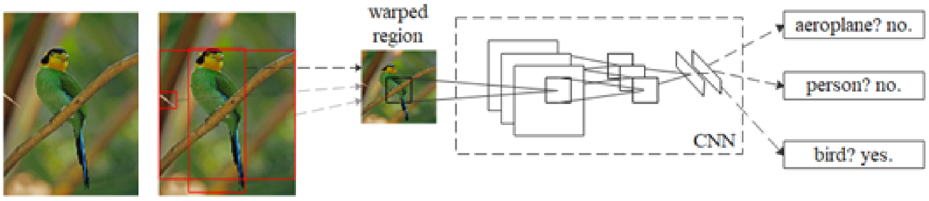


Figure 1: The architecture of R-CNN

algorithms consist of SSD, YOLO, RetinaNet etc. The two-stage detectors contain R-CNN, Fast R-CNN, Faster R-CNN, etc.

2.1 Deep Object Detection Networks

2.1.1 R-CNN. R-CNN (region-based convolution neural network) uses the selective search algorithm to extract about 2000 region proposals in the image, makes them a uniform size and inputs these region proposals into convolutional neutral networks as depicted in Figure 1 [3]. Afterwards, the algorithm makes use of Alexnet to extract features and finally inputs the CNN features of each region proposal into SVM to implement classification with each class corresponding to the SVM classifier. R-CNN has a relatively high accuracy in VOC2007 dataset with mAP(mean average precision) increasing from 34.3% to 66%. The disadvantage of R-CNN is that every proposal requires characteristic extraction which causes a huge amount of calculation, slowing down the training and processing speed and occupying too much disk space. Besides, since all region proposals must be altered into the same size, some of the images become deformed.

2.1.2 SPP-Net. The creation of SPP-Net (Spatial Pyramid Pooling Network) tackles the problem that the fully connected layer requires images of the same size as depicted in Figure 2 [4]. The core variation is adding a Spatial Pyramid Pooling layer between the convolutional layer and the fully connected layer. Instead of extracting features for each proposal of R-CNN, SPP-Net only implements convolution calculation one time for an image. After obtaining the feature map for the entire image, each proposal extracts features directly from the convolution feature of the whole image. Therefore, SPP-Net decreases the amount of calculation remarkably, which speeds up the detection speed. The functionality of SPP layer is to make sure the fully connected layer receives vectors of fixed length.

2.1.3 Fast R-CNN. R-CNN and SPP-Net has some common weaknesses. The first drawback is that they both perform multi-stage training. However, Fast R-CNN implements a single-stage pipeline involving both classification and bounding-box regression using multi-task loss as depicted in Figure 3 [5]. Besides, Fast R-CNN saves training time and storage space as it does not need to store the extracted features to disks as R-CNN and SPP-net do by involving bounding-box regression into CNN network. Fast R-CNN alters the SPP layer in SPP-Net to the ROI(Region of Interesting) pooling layer. ROI pooling layer simplifies the multi-scale pooling of the SPP layer to single-scale pooling which transforms each region of interest to a feature map with the same spatial extent (e.g. 4×4). Then the fixed-size feature map generates a fixed-length vector as the input of the fully-connected layers.

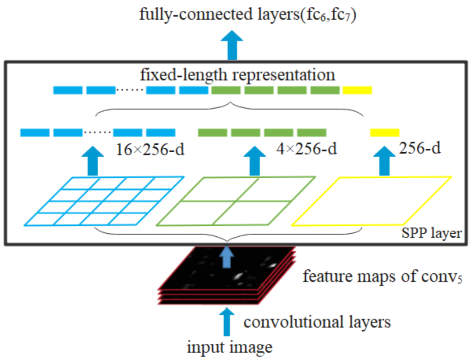


Figure 2: The structure of the SPP layer

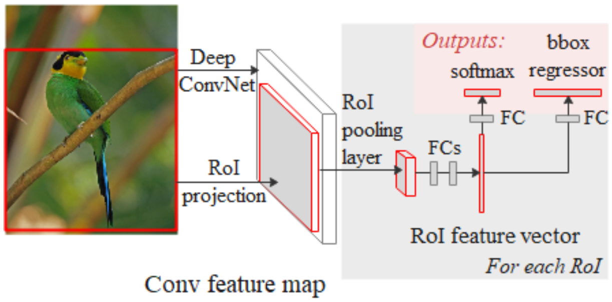


Figure 3: The architecture of Fast R-CNN

2.1.4 Faster R-CNN. Before Faster R-CNN is put forward, computing proposals is the major constraint on the detection speed [6]. The classical method to extract proposals is selective search, used by R-CNN and Fast R-CNN. Faster R-CNN comes up with a new effective way to locate and extract region proposals, that is, by using the RPN (Region proposal network) as depicted in Figure 4. By using RPN, the time consumption of convolutional calculations is reduced remarkably and the detection speed is increased. Faster R-CNN unifies the process of feature extraction, proposal extraction, bounding-box regression, and classification into one neutral network with the help of RPN. Faster R-CNN uses sliding window to generate region proposals of different scales and proportions and uses anchors to predict regions of interest. At each position of the

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

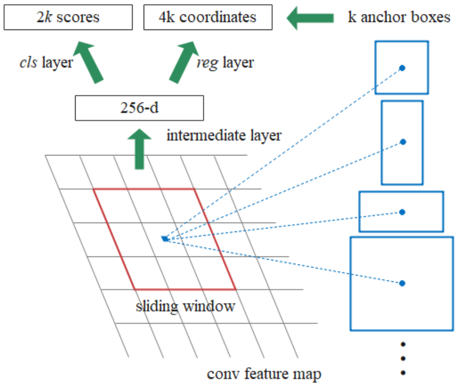


Figure 4: The framework of RPN

feature map, different anchors of fixed scale and proportion are set to predict ROI.

2.1.5 RetinaNet. One-stage detectors have relatively faster detection speed than two-stage object detectors, but with some sacrifice on the accuracy. The lower accuracy of one-stage detectors is caused by class imbalance and to solve this weakness, RetinaNet defines a new focal loss to control the balance between different classes [7]. In one input image, the proportion of background usually is much larger than the proportion of foreground objects which causes that negative examples have too large loss and cover the loss of positive examples. This is not beneficial to the convergence of objective functions. Additionally, many negative examples belong to easy negatives since they are not close to the boundaries of the background and foreground objects and are easy to classify. These easy negatives have low loss which makes the gradient in backward calculation small and limits the convergence of parameters.

By using Focal Loss, the weights of easy and hard examples and the proportion of positive and negative examples can be adjusted reasonably. The formula for Focal Loss is shown below:

$$\boldsymbol{FL}(p_t) = -\alpha_t (1 - p_t)^{\gamma} \log(p_t) \tag{1}$$

p_t denotes the probability for different categories, γ is a constant greater than zero and α_t is a decimal number between 0 and 1. $(1 - p_t)$ makes weights for easy examples low and weights for hard examples high and α_t controls the proportion of background classes and foreground classes.

2.2 Commodity Dataset Preparation

2.2.1 Image Collection. The dataset chosen in this paper is a large-scale product dataset called the RPC (retail product checkout) dataset in the situation of retail product checkout which was released by Kuangshi Nanjing Research Institute [8]. This dataset collects images of products in 17 parent categories and 200 child categories. The training set of RPC consists of 53739 images of one single product placed on a round platform. It collects several images for one product in different shooting angles and shooting positions. While the testing and validation sets simulate real-world product checkout situations with 24000 and 6000 images respectively containing multiple products placed on an empty white platform. According to the complexity, the multi-product images are classified into 3 levels based on the number of products and the number of product categories in the images. The classification criteria are provided in Table 1. The dataset tries best to approach the real situation of product checkout in daily life in the product categories and quantities, the angles of placement and the rates of shelter.

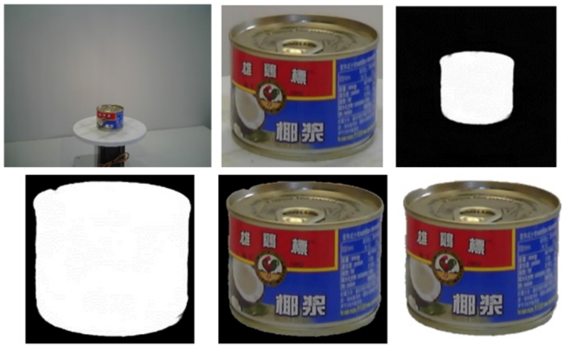


Figure 5: The results of image segmentation

The original training set provided by the RPC dataset only contains one product in each image and these images need to be processed to obtain similar images in the testing or validation sets with several products placed on the checkout platform in one image. The product image processing includes segmentation, synthesis and rendering.

2.2.2 Product Image Segmentation. The purpose of segmentation is separating and extracting each product from the images in the RPC dataset. First the single-product images are annotated and cut using single bounding box and double bounding box. And then the double bounding box product images are processed to separate the products from the background through Salient Object Detection [9]. Next, the results and the single bounding box images perform CRF refinement and the synthesized images are cut to the size of single bounding box to get the masks of the products. And then the masks are used to separate the product images from the single bounding box images. Finally, the black backgrounds of the product images are transformed to become transparent and the images of each separated product are obtained in Figure 5

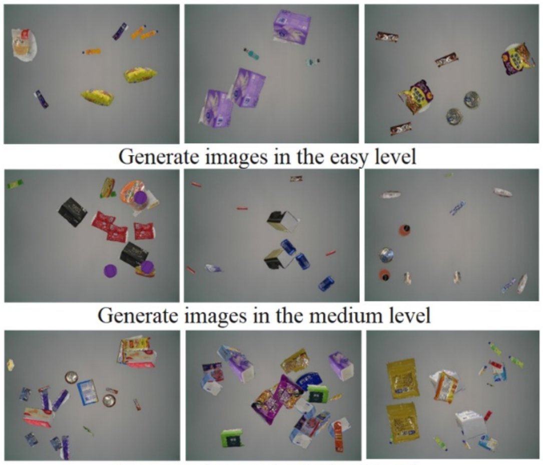
2.2.3 Product Image Synthesis. After obtaining the images of single products, these images are copied and randomly pasted to the background images which are the same as the backgrounds in the testing and validation sets. Based on the criteria for the three levels of images, the single-product images are pasted to the backgrounds with different numbers and categories of products to generate the three levels of images in Figure 6. To simulate real situations of product checkout, positions and shooting angles of the pasted products are both at random and the rate of coverage between any two products is less than 50%.

2.2.4 Product Image Rendering. Finally, CycleGAN [10] is used to render the multi-product images to add lighting effects in Figure 7, which tries best to simulate real checkout situations.

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

Table 1: The classification criteria for the of images

Clutter levels	Number of categories	Number of instances
Easy	3-5	3-10
Medium	5-8	10-15
Hard	8-10	15-20



Generate images in the easy level

Generate images in the medium level

Generate images in the hard level

Figure 6: The results of image synthesis

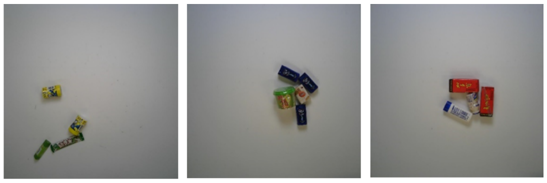


Figure 7: The effects of the images rendering

2.2.5 Flow Path of Commodity Recognition Based on Deep Networks. To improve the efficiency of retail product checkout and decrease labor costs, this paper designs a commodity recognition method based on object detection networks which consists of the following three steps:

- Step 1: Image collection. Collect images in the situation of real-world retail product checkout. Use cameras to take pictures of several products from different shooting angles to prepare for the production of the train set and also take pictures with multiple products placed on an empty platform to generate the validation set.
- Step 2: Train set production. Segment single products from the collected images and paste multiple products to the empty background image to synthesis images containing multiple products on the platform, simulating real-world retail product checkout situations. These synthesized images will be rendered to add lightening effects and then make up for the train set.
- Step 3: Network training and selection. Train different object detection networks on the prepared train set, compare the detection results for each network on the validation set and analyze their performance to select better ones to apply to real-world commodity recognition tasks.

3 EXPERIMENTS

3.1 Experimental Environments

The experiment is performed under Linux operating system and uses the deep learning framework PyTorch. In addition, NVIDIA GPU is used to accelerate the processing speed and training speeding of the networks. The detailed hardware parameters are listed in Table 2

3.2 Experiment Results

Taking the real-world commodity recognition situations into account, both accuracy and speed should be considered when choosing object detection networks to satisfy the requirements that retail product checkout should be performed as fast as possible with enough accuracy. Therefore, we choose Faster R-CNN, YOLOv3 [11] and RetinaNet. The results in Table 3 show that Faster R-CNN has a higher accuracy on the MS COCO dataset than Fast R-CNN and YOLOv3 as depicted in Figure 8, its speed exceeds the preceding algorithms like R-CNN, SPP-net and Fast R-CNN. RetinaNet combines the advantages of one-stage object detectors and two-stage object detectors, considering both accuracy and detection speed.

We also perform the ACO (Automatic Check-out) task which simulates the real-world retail checkout situation. The results are listed in Table 4. cAcc denotes the checkout accuracy of the entire commodity list, which means that the checkout task is considered successful if and only if all the products on the image are classified and located correctly. This value is quite essential and practical since in real-world checkout situations, it is important to recognize all products on the platform correctly. Data in Table 4 shows that for RetinaNet, 91.65% of images in the easy level completes the ACO task successfully, that is, all products on each image are classified correctly. And as the complexity of the images increases, with more products and more categories, the overall accuracy reduces. Except from the increased complexity, this may result from the fact that with more products, the distances between them decrease and it is more difficult for the detection networks to predict denser objects. Additionally, ACD denotes the mean number of counting errors for each image. Table 4 shows that the errors become more for more difficult levels.

From the accuracy of the three object detection networks we can see that Faster R-CNN obtains the highest accuracy of the three algorithms which is 89.21% and RetinaNet also performs

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

Table 2: The parameters of experimental environments

Name	Parameter Specification
NVIDIA GPU	Nvidia Tesla P100
Internal storage	30GB
Processor	Intel Xeon E5-2682v4, 2.5 GHz, 4 CPUs, 8 Threads,

Table 3: The detection results of the three deep networks

Algorithm	Backbone	AP	AP50	AP75
RetinaNet	ResNet-101-FPN	89.03%	99.49%	99.56%
Faster R-CNN	ResNet-101-FPN	89.21%	95.45%	96.98%
YOLOv3	Darknet-53	75.02%	99.65%	82.32%

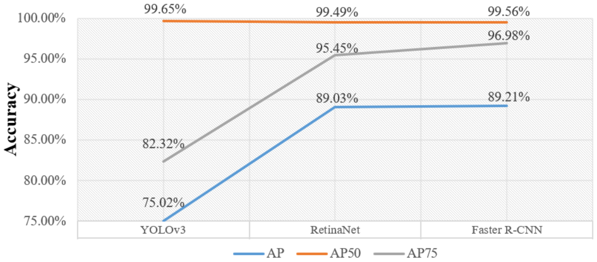


Figure 8: The accuracy of deep networks

well, with 89.03% mAP value. YOLOv3 is faster than the other two algorithms on the RPC dataset but its AP value is much lower. However, YOLOv3 has a great performance on AP50 when IoU (intersection over union) equals to 0.50 and with IoU increasing, its detection accuracy deduces. From this we can infer that YOLOv3 is better than the other two at predicting small objects but poor at predicting large objects.

The overall detection accuracy of object detection algorithms still has some potentials to increase. Firstly, the placement angles and shooting angles of products are heterogeneous and variable. Customers place products on the platform randomly and each product can generate various forms and appearances in the images. This adds plenty of difficulties to the data collection for the dataset and the dataset should be large enough to involve all forms of each product with each form appearing in the dataset a great many times. Now the training set only contains 30000 images and by enlarging the size of the dataset, the accuracy is possible to increase. Besides, from the mAP values of each product category we find that some categories have much lower accuracies than others which decreases the average accuracy of all categories. This is because when synthesizing images for the train set, the products are chosen to be pasted to the background completely at random, which makes some commodity categories not pasted enough times and are fewer than others. By changing the image synthesis strategy to make sure the images contain enough amounts of each product, the accuracy will possibly become higher.

4 CONCLUSIONS

In conclusion, we propose a deep learning framework of object detection for commodity image recognition. Firstly, the commodity image dataset is constructed. Collect images in the situation of real-world retail product checkout. Segment single products from the collected images, synthesis and render add the images to make up for the train set. and paste multiple products to the empty background image to synthesis images containing multiple products on the platform, simulating real-world retail product checkout situations. These synthesized images will be rendered to add lightening effects and then make up for the train set. Then, the image data is trained via object detection deep networks. We compare the detection results of the deep networks and analyze their performance. The results show that the deep learning methods have a great perspective to commodity image recognition.

ACKNOWLEDGMENTS

This work was supported by the Open Grant funded by Zhejiang Key Laboratory of Equipment Electronics Research in Hangzhou Dianzi University (GK190901207001).

REFERENCES

[1] Rui Chen, Meiling Wang, Yi Lai. 2020. Analysis of the Role and Robustness of Artificial Intelligence in Commodity Image Recognition under Deep Learning

Table 4: The results for different levels of images on the ACO task of RetinaNet

Level	cAcc	ACD
Easy	91.65%	0.12
Medium	82.3%	0.20
Hard	71.65%	0.41

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

Neural Network. PLOS ONE, 15, 7, (Jul. 2020), 1–17. https://doi.org/10.1371/journal.pone.0235783

[2] Xiaofeng Zou, Liqian Zhou, Kenli Li, *et al.* 2020. Multi-Task Cascade Deep Convolutional Neural Networks for Large-Scale Commodity Recognition. Neural Computing and Applications, 32, 10, (Oct. 2020), 5633–5647. https://doi.org/10.1007/s00521-019-04311-9

[3] Ross Girshick, Jeff Donahue, Trevor Darrell, *et al.* 2014. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR2014). IEEE, New York, NY, 580–587. https://doi.org/10.1109/CVPR.2014.81

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, *et al.* 2015. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 37, 9, (Sep. 2015), 1904–1916.. https://doi.org/10. 1109/TPAMI.2015.2389824

[5] Girshick, Ross. 2015. Fast R-CNN. Proceedings of the 2015 IEEE Conference on Computer Vision (Santiago, Chile. Dec. 7-13, 2015). (ICCV2015). IEEE, New York, NY, 1440–1448. https://doi.org/10.1109/ICCV.2015.169

[6] Shaoqing Ren, Kaiming He,Ross Girshick, *et al.* 2017. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. 39, 6, (Jun. 2017), 1137–1149. https://doi.org/ 10.1109/TPAMI.2016.2577031

[7] Tsung-Yi Lin,Priya Goyal,Ross Girshick, *et al.* 2020. Focal Loss for Dense Object Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence. 42, 2, (Feb. 2020), 318–327. https://doi.org/10.1109/TPAMI.2018.2858826

[8] Xiu-Shen Wei, Quan Cui, Lei Yang, *et al.* 2019. RPC: A Large-Scale Retail Product Checkout Dataset. ArXiv Preprint ArXiv:1901.07249. from https://arxiv.org/abs/1901.07249

[9] Ping Hu, Weiqiang Wang, Chi Zhang, *et al.* 2016. Detecting Salient Objects via Color and Texture Compactness Hypotheses. IEEE Transactions on Image Processing. 25, 10, (Oct. 2016), 4653–4664. https://doi.org/10.1109/TIP.2016.2594489

[10] Jun-Yan Zhu,Taesung Park,Phillip Isola, *et al.* 2017. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV2017), IEEE, New York, NY, 2242–2251. https://doi.org/10.1109/ICCV.2017.244

[11] Sichkar, V. N., and S. A. Kolyubin. 2020. Real Time Detection and Classification of Traffic Signs Based on YOLO Version 3 Algorithm. Scientific and Technical Journal of Information Technologies, Mechanics and Optics. 20, 3, (Mar. 2016), 418–424. https://doi.org/ 10.17586/2226-1494-2020-20-3-418-424

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net