A Robust Prototype-free Retrieval Method for Automatic Check-Out

Huijie Huangfu
College of Computer Science
Sichuan University
Chengdu, China
huangfu@stu.scu.edu.cn

Ziyuan Yang
College of Computer Science
Sichuan University
Chengdu, China
yangziyuan@stu.scu.edu.cn

Maosong Ran
College of Computer Science
Sichuan University
Chengdu, China
maosongran@gmail.com

Weihua Zhang
College of Computer Science
Sichuan University
Chengdu, China
zhangweihua@scu.edu.cn

Jingfeng Lu
College of Computer Science
Sichuan University
Chengdu, China
jingfeng.lu@scu.edu.cn

Yi Zhang
School of Cyber Science and Engineering
Sichuan University
Chengdu, China
yzhang@scu.edu.cn (Corresponding Author)

2023 IEEE Symposium on Computers and Communications (ISCC) | 979-8-3503-0048-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISCC58397.2023.10218102

***Abstract*—In recent years, automatic check-out (ACO) gains increasing interest and has been widely used in daily life. However, current works mainly rely on both counter and product prototype images in the training phase, and it is hard to maintain the performance in an incremental setting. To deal with this problem, we propose a *robust prototype-free retrieval method (ROPREM)* for ACO, which is a cascaded framework composed of a product detector module and a product retrieval module. We use the product detector module without product class information to locate products. Additionally, we first attempt the check-out process as a retrieval process rather than a classification process. The retrieval result is considered as the product class by comparing the feature similarity between a query image and gallery templates. As a result, our method require much fewer training samples and achieves state-of-the-art performance on the public Retail Product Checkout (RPC) dataset.**

***Index Terms*—Automatic check-out, object detection, object retrieval, prototype-free**

I. Introduction

Recently, automatic check-out (ACO) systems have been increasingly used in vending machines and supermarkets. Using the ACO system can quickly know the category and quantity of products selected by the user, thereby reducing labor costs and speeding up checkout. As a key component of ACO system, computer vision (CV)-based ACO method is user-friendly, low-cost, and easily maintainable, which has become the mainstream in ACO systems.

This kind of methods aims to count all product categories based on the counter image [1]. The counter image is a picture of the customer displaying all the selected products on the counter. They usually use the outputs of object detection methods to count the products directly. However, usually there are many categories of products. So training an object detection network to detect hundreds of categories requires a large amount of well-annotated samples, which are time-consuming and labor-intensive in practice. To alleviate this problem, researchers proposed to introduce product prototype images to generate numerous synthesized images [1]. However, The acquisition and annotation of product prototype images increase the human labor. Additionally, since the appearances of a product from different viewing angles are significantly different, it is impossible to enumerate all combinations of different views of different products to produce a comprehensive prototype image dataset. Moreover, the domain gap between real and synthesized images may lead to a performance drop. Another shortcoming of this method lies in the lack of flexibility and high maintenance cost in practice. Since the item categories are often variable and the number of output categories of the detection network is usually fixed, we have to fine-tune or retrain the model to adapt new categories.

Another kind of ACO methods is regression-based. Product prototype images accompanied by a counter image are sent to the network to predict a density map [6]. However, the performance of these methods highly relies on the quality of the prototype images. It is expected that more reference images randomly chosen from the product prototype images can boost the performance, but due to hardware limitations, only a small number of reference images can be used in one inference. Meanwhile, since this kind of methods only infers one category at a time, the time overhead is unbearable when thousands of product categories exist in practice.

To handle the above problems, in this paper, we propose a novel ***ro**bust **p**rototype-**f**ree re**t**rieval **m**ethod (**ROPREM**)* for ACO. Instead of using product prototype images to generate simulated counter images and assist in product location, we design a cascaded check-out framework based on object detection and product retrieval. Specifically, the proposed framework includes two components, a product detector module and a product retrieval module. Compared with classification networks which are trained on known classes, there is no predefined class in our framework, and a query image may come from an item from an unseen class [19]. In automatic counting, since the number of categories is often variable,

retrieval is more flexible than classification. In the training phase, the detector and retrieval modules are trained separately only with the counter images. In the testing phase, given a test counter image, we use the detector to obtain the product locations as the query set and the cropped product images from the training set are treated as the gallery set. The product classes are obtained in our retrieval module by measuring the similarity between the features extracted from the query set and gallery set. The main contributions of this work are three-fold:

- We develop a novel prototype-free retrieval method for ACO. Different from other models trained with both counter and prototype images, the proposed ROPREM is trained only with the counter images, which efficiently avoids the domain gap between the real and synthesized datasets.
- To our best knowledge, this is the first retrieval-based ACO method. The proposed product retrieval module enables our method to handle unseen products in an incremental setting.
- Extensive experiments are conducted to validate our method on the public dataset and the results demonstrate ROPREM achieves competitive performance in comparison with SOTA methods in both traditional and incremental settings.

II. Related Works

A. Automatic Check-Out

With the rapid development of artificial intelligence, ACO has attracted more and more attention. Product counting is one of the most important tasks in ACO.

Object detection methods such as Faster-RCNN [13] are employed for check-out by detecting and identifying all item categories in the counter image. However, it is time-consuming and labor-intensive to acquire massive well-annotated counter images with precise bounding box annotations for these methods in practice. As a result, how to maintain the performance with limited samples is the key point. In order to collect enough training samples, Retail Product Checkout (RPC) [1] uses product prototype images from different angles and adopts a synthesize-and-render strategy to generate abundant realistic images to train a product detector. Inspired by this method, Li *et al.* [4] employed the pose pruning method to optimize image generation. Moreover, they proposed the data priming network (DPNet) to select reliable testing data to relieve the domain gap problem. Yang *et al.* [5] proposed IncreACO with Photorealistic Exemplar Augmentation (PEA), which further improves the reality of the generated check-out images and adopts teacher-student pattern to resolve the absence of labeled check-out data. Hao *et al.* [7] proposed prototype-based classifier learning from Single-Product Exemplars (PSP). By training a prototype-based classifier with the product prototype images, they can distinguish the detection proposals of trained classes and backgrounds to improve the model.

Another main direction for ACO is density map prediction. Hao *et al.* [6] proposed a self-supervised multi-category counting network (S2MC2) to generate the counts of a product directly aided by the product prototype images. The feature map of the counter image and averaged feature map of the product prototype images are concatenated to feed into an hourglass network to generate a density map. The final counting number is calculated from the predicted density map.

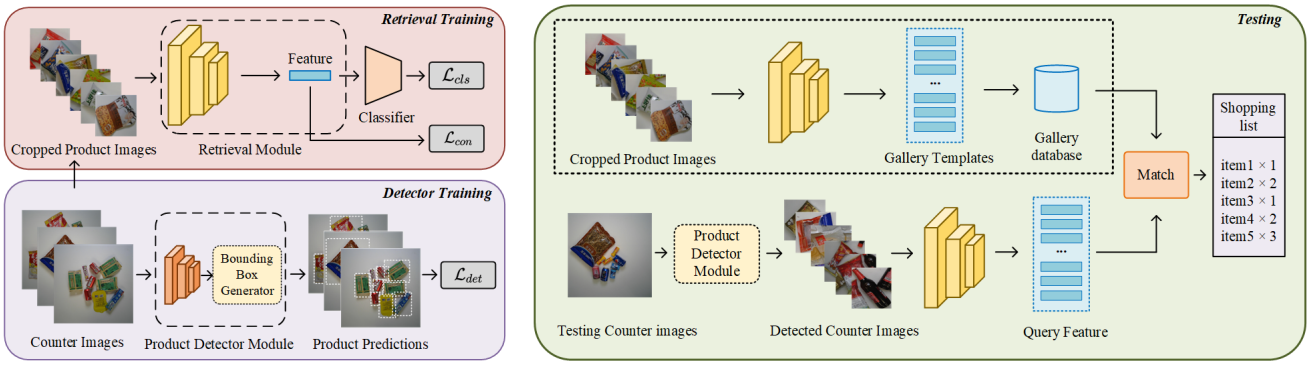
B. Content-Based Image Retrieval

Contend-Based Image Retrieval (CBIR) aims to retrieve related images from an image gallery with query image, which has been widely used in many tasks, including person re-identification, vehicle re-identification, landmark retrieval, remote sensing, and online product searching [24]. In CBIR, there are two important components, image feature representation and similarity measure. Image feature representation is expected to generate discriminative and robust feature vectors. The similarity measure is utilized to measure the similarity of feature vectors extracted from different objects in feature space. Recently, researchers introduced deep metric learning to enforce intra-class discrimination and gain more fine-grained instance-level image representations. Radenovic *et al.* [25] introduced contrastive loss with negative and positive examples to enhance the image representation. Cheng *et al.* [22] presented a multi-channel parts-based convolutional network for person re-identification problem, which is formulated under a triplet framework. However, these works start from predefined bounding boxes, but in real application scenarios, it needs to be combined with a detection network. Zheng *et al.* [23] combined pedestrian detection and person recognition to design an end-to-end re-ID system based on large-scale datasets.

III. Methodology

A. Overall Framework

In previous works [4] [5] [7], a detector with predefined categories is usually used. This architecture needs more training data and has to fine-tune the model for new categories. In order to alleviate this problem, the proposed ROPREM splits the ACO procedure into two steps, product detection and product retrieval, which correspond to the product detector and product retrieval modules, respectively. The basic concept of ROPREM is illustrated in Fig 1. In the training phase, the product detector module is trained to locate products in the counter images without product class information. Then, the retrieval module is trained with cropped product images to obtain discriminative features. In the testing phase, the product detector locates the products from the input counter image and the detected products are cropped into separate product images. Then, we utilize the retrieval module to extract the features and match them with the gallery template dataset, which is built using training samples. Finally, the class of TOP-1 feature similarity is treated as the retrieval result, and the counting results are the sum of retrieval results in a counter image.



(a) (b)

Fig. 1. Overview of our method. 1(a): training phase of product detector module and retrieval module. 1(b): testing phase of whole module.

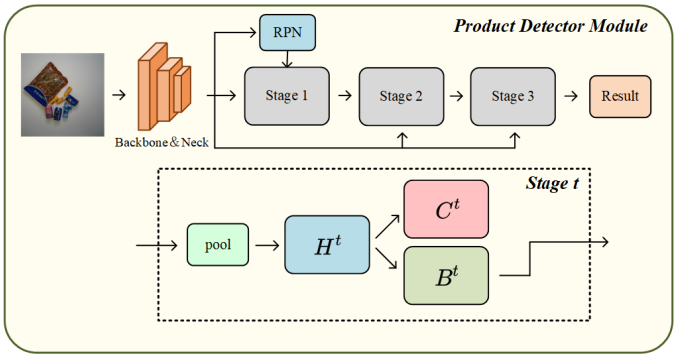


Fig. 2. Structure of product detector module

B. Product Detector Module

The structure of the product detector module is shown in Fig 2. A cascaded R-CNN [16] is employed as our product detector, which contains three main parts: backbone network, region proposal network (RPN) and region of interest network. Specifically, we adopt ResNet-50 [18] as the backbone network. Then, the feature pyramid network (FPN) [14] is connected to the backbone network as the neck network. With top-down architecture and lateral connections, FPN can fuse high-level features with low-level features to obtain high-resolution features with strong semantic information. There are several stages in the cascaded R-CNN. Instead of using a single detection head which generates low quality proposals, a cascaded multi-stage R-CNN network is used to improve the proposal with lower intersection over union (IoU) gradually. The cascaded specialized regressor is provided as:

$$f(x,b) = f_T \circ f_{T-1} \circ \cdots \circ f_1(x, b_0), \tag{1}$$

where x is the image patch, b denotes the estimated bounding box, and T represents the number of cascade stages. Initial bounding box b_0 is generated by RPN. f_t is optimized from b_{t-1} for stage $t \in \{1, ..., T\}$.

In product detector, IoU is used to define the positive and negative proposals in the training phase. For one R-CNN detector, a low IoU threshold may degrade the detection accuracy. However, simply using a high IoU threshold will decrease the number of positive samples and cause overfitting problem [16]. In order to solve the above problems, we use a cascaded structure to increase the IoU gradually and guarantee the number of positive samples. This process can be seen as a re-sampling process. As a result, we can improve the estimation accuracy and stabilize the number of positive proposals. Particularly, we select a specific IoU threshold set for each stage in our method. In this paper, our detector contains three stages and the threshold set is empirically set to $0.5, 0.6, 0.7$.

Stage t contains one detection head H^t, one classifier C^t and one bounding box regressor B^t with IoU threshold u^t. Then, the loss function is defined as:

$$L(x^t, g) = L_{\text{cls}}(C(x^t), y^t) + \lambda_1 [y^t \geq 1] L_{\text{loc}}(B(x^t, b^t), g), \tag{2}$$

where b^t is the estimated bounding box obtained from the previous stage, x^t and g are the proposal image and its ground truth bounding box. y^t is the predicted label of x^t where $IoU(x^t, g) \geq u^t$. $y^t = 0$ when the proposal is considered as background. $[\cdot]$ stands for the indicator function. λ_1 is the trade-off coefficient. L_{cls} and L_{loc} are the cross-entropy loss and L_1 loss, respectively. Each stage adjusts the bounding boxes to higher IoU for the next stage. Then, the proposal quality is gradually improved with more stages.

In this module, each product is annotated as only one class "product" rather than a detailed product category. There are two main benefits: 1) Training a single-category detector is easier than training a multi-category detector. Hence, we don't need to synthesize counter images and fewer samples are required to train the product detector. 2) Training a multi-category detector makes the network inflexible and we have to fine-tune the model to adapt to new categories. Our method belongs to a simple binary classification, which enables our model compatible with incremental setting.

C. Retrieval Module

Since the categories are often variable in ACO, the number of classes in traditional classification network is fixed and has

TABLE I
EXPERIMENTAL RESULTS ON RPC DATASET

Clutter mode	Methods	Prototype image	cAcc ↑	ACD ↓	mCCD ↓	mCIoU ↑
Easy	Wei *et al.*	-	73.17%	0.49	0.07	93.66%
	IncreACO	-	88.06%	0.21	0.03	96.95%
	DPNet	-	90.32%	0.10	0.02	97.87%
	S2MC2	-	90.03%	0.25	0.02	98.22%
	PSP	-	92.12%	0.11	0.02	98.40%
	ROPREM(ours)	-	**94.15%**	**0.10**	**0.01**	**98.57%**
Medium	Wei *et al.*	-	54.69%	0.90	0.08	92.95%
	IncreACO	-	77.31%	0.40	0.03	96.82%
	DPNet	-	80.68%	0.32	0.03	97.38%
	S2MC2	-	83.66%	0.46	0.02	98.15%
	PSP	-	87.28%	0.18	0.01	98.59%
	ROPREM(ours)	-	**92.82%**	**0.11**	**0.01**	**99.12%**
Hard	Wei *et al.*	-	42.48%	1.28	0.07	93.06%
	IncreACO	-	66.14%	0.64	0.04	93.35%
	DPNet	-	70.76%	0.53	0.03	97.04%
	S2MC2	-	72.75%	0.79	0.02	97.80%
	PSP	-	80.67%	0.29	0.02	98.29%
	ROPREM(ours)	-	**87.79%**	**0.19**	**0.01**	**98.87%**
Averaged	Wei *et al.*	53739	56.68%	0.89	0.07	93.19%
	IncreACO	53739	77.15%	0.41	0.03	96.72%
	DPNet	53739	80.51%	0.34	0.03	97.33%
	S2MC2	53739	82.12%	0.51	0.02	97.96%
	PSP	1600	86.69%	0.19	0.02	98.44%
	ROPREM(ours)	0	**91.59%**	**0.13**	**0.01**	**98.92%**

high maintenance cost to update the model in practice. As a result, we subdivide the classification operation into two steps, product retrieval and product counting. To this end, we propose a winner-take-all mechanism-based retrieval module, which treats the class from the most similar gallery feature as the result. Given a gallery set $\mathcal{G} = \{g_i\}_{i\in N}^{N}$ where g_i is the gallery image from the product images cropped from the counter image dataset, for a single query image q, its label p is predicted as:

$$p = \mathrm{argmax}\mathbb{E}_{g_i\sim\mathcal{G}}(\mathrm{sim}(f(q), f(g_i))), \tag{3}$$

where $\mathrm{sim}(\cdot,\cdot)$ is the similarity function. $f(\cdot)$ denotes the feature extractor and in this paper, we use ResNet-50 for feature extraction. In addition, to make the extracted features more discriminative, we use cross-entropy loss L_{CE} and contrastive loss L_{con}, and the total loss is formulated as:

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda_2\mathcal{L}_{con}, \tag{4}$$

where λ_2 is the trade-off coefficient. The contrastive loss L_{con} is defined as:

$$\begin{aligned}\mathcal{L}_{con}(f_i, f_j) = [y_i = y_j]\,\|f_i - f_j\|_2^2 + \\ [y_i \neq y_j]\max\left(0, m - \|f_i - f_j\|_2\right)^2,\end{aligned} \tag{5}$$

where f_i and f_j denote the feature representation of product images pair. y_i and y_j are the corresponding labels. m is the distance threshold for two different categories.

In the training phase, we use cropped product images in the training set to train a feature extractor. After that, we collect the features of these images as gallery feature templates to set up a feature database. In the testing stage, we first use the feature extractor to get the query feature from the query image. Then, cosine similarity is adopted to measure the similarity between the query feature f_q and gallery feature templates f_g in the database. The cosine similarity is defined as:

$$\mathrm{sim}(f_q, f_g) = \frac{f_q \cdot f_g}{\| f_q \| \times \| f_g \|}. \tag{6}$$

Since our product detector and retrieval modules are independent, they can be trained in parallel and the training time overhead is efficiently reduced. In addition, as mentioned above, another advantage of our method is prototype free. Only the counter image dataset is used to train both product detector and retrieval modules and no product prototype image is included. For ACO, retrieval is more flexible than classification in an incremental setting. In image classification training, all classes are known and the network easily overfits for the predefined classes.

IV. EXPERIMENTS

A. Implementation Details

Our ROPREM is implemented with PyTorch on a PC (AMD Ryzen 7 5800X, a Nvidia RTX 3080ti GPU and 32G memory). The size of counter image is 800×800, and 256×256 for cropped product image. The batch size is set to 4. Adam [26] optimizer is employed and the learning rate is set to 1×10^{-4}. λ_1 and λ_2 are set to 1 for trade-off.

B. Dataset

We use the public RPC dataset to evaluate our proposed ROPREM. In RPC, there are 200 product categories in total and the dataset contains three parts. The training set contains 53739 images of product prototype images belonging to 200 categories. The validation set contains counter images taken from the checkout system. There are 6000 images with 13 objects and 6 categories per image. The test set has the same

setting and size as the validation set, and there are 24000 images with 12 objects and 6 categories per image. In the validation and test sets, there are three types of clutters: easy (3-5 categories and 3-10 instances), medium (5-8 categories and 10-15 instances), and hard (8-10 categories and 15-20 instances) modes. In this work, since the proposed model does not need the product prototype images, the original training set is abandoned and the validation set is adopted as the training set. For incremental settings, we follow the setup in [6] that 17 categories are chosen as the new categories. Four metrics, including check-out accuracy (cAcc), average counting distance (ACD), mean category counting distance (mCCD), and mean category intersection of union (mCIoU), are employed to quantitatively evaluate the proposed method.

C. Results

To validate the performance of the proposed method, we compare our method with several state-of-the-art methods, including Wei et al. [1], IncreACO [5], DPNet [4], PSP [7] , and S2MC2 [6]. The results are listed in Table I. As shown in Table I, it can be seen that our method achieves the best performance in terms of all the metrics for different modes. Compared with other methods aided by the synthesize-and-render strategy, we achieved 11.08% higher cAcc than DPNet. Since the appearances of a product from different angles are quite different, synthesizing counter images by enumerating all the combinations of different products is infeasible. In addition, there is a huge domain gap between the counter image and the product prototype image, which may degrade the performance. It is worth mentioning that our method is the only prototype-free method in these methods and we still achieve the best scores without the help of product prototype images. In [1] [4] [5], 53739 product prototype images are used to synthesize counter images. In [6], 53739 product prototype images are utilized to regress the product position. In [7], the authors use 1600 product prototype images to train a classifier. We achieve 2.03%, 5.54%, 7.12%, 4.90% higher cAcc than PSP in easy, medium, hard, and averaged modes, respectively. The result supports the effectiveness of our approach.

D. Incremental Study

To verify the performance of our method for incremental setting, we follow the experimental setup in [6] that 17 categories are chosen as the new categories. The remaining 183 categories are used for training. As a result, the number of training images decreases from 6000 to 3324. The cAcc values of different methods and experimental setups are illustrated in Fig 3. ”183/200” means there are 183 categories for training and 200 categories for testing. ”183/183” means there are 183 categories in both training and testing sets. Other methods directly classify the products. Our retrieval method is flexible for unseen categories compared with classification. Since there are no predefined categories in retrieval, we don’t need to fine-tune the model for new categories. It can be seen that our proposed method achieves the best performance

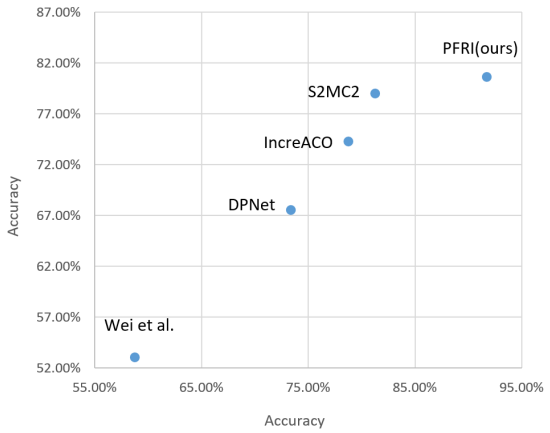


Fig. 3. Experimental results in the incremental setting. The x-axis represents accuracy in ”183/183” mode, and the y-axis represents accuracy in ”183/200” mode.

TABLE II
ABLATION STUDY ON CONTRASTIVE LOSS

Contrastive loss	cAcc ↑	ACD ↓	mCCD ↓	mCIoU ↑
×	88.75%	0.18	0.01	98.59%
√	91.59%	0.13	0.01	98.92%

in both ”183/183” and ”183/200” cases. Compared with the previous method S2MC2 [6], the cAcc is 10.44% higher in the ”183/183” mode and 1.66% higher in the ”183/200” mode, respectively.

E. Ablation Study

In this part, we conduct ablation studies to sense the impact of contrastive loss on our module. Two versions of the retrieval module, with and without contrastive loss are trained. As shown in Table II, the cAcc decreases by 2.84%, when we remove the contrastive loss. The results support that contrastive loss improves the feature representations in the retrieval module. By measuring the distance between different features, more discriminative features are extracted using contrastive loss.

V. CONCLUSIONS

In this paper, we propose a prototype-free retrieval incremental method (ROPREM) for ACO. We use a product detector to obtain product locations and a product retrieval module to retrieve detected images from gallery templates. Different from existing methods, our method is prototype-free and incremental. We achieve 91.59% cAcc without the help of a large number of product prototype images. ROPREM can achieve the best performance in comparison with other SOTA methods in traditional and incremental settings. However, our method still face some challenges, for example, multi-category labels are necessary in our training process, which is labor-intensive and impractical to continually label emerging new products. In future works, we will explore how to reduce the reliance on a large amount of well-annotated labels.

REFERENCES

[1] Wei, X., Cui, Q., Yang, L., Wang, P., Liu, L. & Yang, J. RPC: a large-scale and fine-grained retail product checkout dataset. (Science China Press,2022)

[2] Bocanegra, C., Khojastepour, M., Arslan, M., Chai, E., Rangarajan, S. & Chowdhury, K. RFGo: a seamless self-checkout system for apparel stores using RFID. *Proceedings Of The 26th Annual International Conference On Mobile Computing And Networking*. pp. 1-14 (2020)

[3] Athauda, T., Marin, J., Lee, J. & Karmakar, N. Robust low-cost passive UHF RFID based smart shopping trolley. *IEEE Journal Of Radio Frequency Identification*. **2**, 134-143 (2018)

[4] Li, C., Du, D., Zhang, L., Luo, T., Wu, Y., Tian, Q., Wen, L. & Lyu, S. Data priming network for automatic check-out. *Proceedings Of The 27th ACM International Conference On Multimedia*. pp. 2152-2160 (2019)

[5] Yang, Y., Sheng, L., Jiang, X., Wang, H., Xu, D. & Cao, X. Increaco: incrementally learned automatic check-out with photorealistic exemplar augmentation. *Proceedings Of The IEEE/CVF Winter Conference On Applications Of Computer Vision*. pp. 626-634 (2021)

[6] Chen, H., Zhou, Y., Li, J., Wei, X. & Xiao, L. Self-Supervised Multi-Category Counting Networks for Automatic Check-Out. *IEEE Transactions On Image Processing*. **31** pp. 3004-3016 (2022)

[7] Chen, H., Wei, X., Zhang, F., Shen, Y., Xu, H. & Xiao, L. Automatic Check-Out via Prototype-Based Classifier Learning from Single-Product Exemplars. *European Conference On Computer Vision*. pp. 277-293 (2022)

[8] Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. You only look once: Unified, real-time object detection. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*. pp. 779-788 (2016)

[9] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. & Berg, A. Ssd: Single shot multibox detector. *European Conference On Computer Vision*. pp. 21-37 (2016)

[10] Lin, T., Goyal, P., Girshick, R., He, K. & Dollár, P. Focal Loss for Dense Object Detection. *2017 IEEE International Conference On Computer Vision (ICCV)*. pp. 2999-3007 (2017)

[11] Zhu, C., He, Y. & Savvides, M. Feature Selective Anchor-Free Module for Single-Shot Object Detection. *2019 IEEE/CVF Conference On Computer Vision And Pattern Recognition (CVPR)*. pp. 840-849 (2019)

[12] Ghiasi, G., Lin, T. & Le, Q. NAS-FPN: Learning Scalable Feature Pyramid Architecture for Object Detection. *2019 IEEE/CVF Conference On Computer Vision And Pattern Recognition (CVPR)*. pp. 7029-7038 (2019)

[13] Ren, S., He, K., Girshick, R. & Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances In Neural Information Processing Systems*. **28** (2015)

[14] Lin, T., Dollár, P., Girshick, R., He, K., Hariharan, B. & Belongie, S. Feature Pyramid Networks for Object Detection. *2017 IEEE Conference On Computer Vision And Pattern Recognition (CVPR)*. pp. 936-944 (2017)

[15] He, K., Gkioxari, G., Dollár, P. & Girshick, R. Mask R-CNN. *2017 IEEE International Conference On Computer Vision (ICCV)*. pp. 2980-2988 (2017)

[16] Cai, Z. & Vasconcelos, N. Cascade R-CNN: Delving Into High Quality Object Detection. *2018 IEEE/CVF Conference On Computer Vision And Pattern Recognition*. pp. 6154-6162 (2018)

[17] Pang, J., Chen, K., Shi, J., Feng, H., Ouyang, W. & Lin, D. Libra R-CNN: Towards Balanced Learning for Object Detection. *2019 IEEE/CVF Conference On Computer Vision And Pattern Recognition (CVPR)*. pp. 821-830 (2019)

[18] He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*. pp. 770-778 (2016)

[19] Zheng, L., Yang, Y. & Hauptmann, A. Person re-identification: Past, present and future. *ArXiv Preprint ArXiv:1610.02984*. (2016)

[20] Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., Zhang, Z., Cheng, D., Zhu, C., Cheng, T., Zhao, Q., Li, B., Lu, X., Zhu, R., Wu, Y., Dai, J., Wang, J., Shi, J., Ouyang, W., Loy, C. & Lin, D. MMDetection: Open MMLab Detection Toolbox and Benchmark. *ArXiv Preprint ArXiv:1906.07155*. (2019)

[21] Wei, Y., Tran, S., Xu, S., Kang, B. & Springer, M. Deep learning for retail product recognition: Challenges and techniques. *Computational Intelligence And Neuroscience*. **2020** (2020)

[22] Cheng, D., Gong, Y., Zhou, S., Wang, J. & Zheng, N. Person Re-identification by Multi-Channel Parts-Based CNN with Improved Triplet Loss Function. *2016 IEEE Conference On Computer Vision And Pattern Recognition (CVPR)*. pp. 1335-1344 (2016)

[23] Zheng, L., Zhang, H., Sun, S., Chandraker, M., Yang, Y. & Tian, Q. Person Re-identification in the Wild. *2017 IEEE Conference On Computer Vision And Pattern Recognition (CVPR)*. pp. 3346-3355 (2017)

[24] Li, X., Yang, J. & Ma, J. Recent developments of content-based image retrieval (CBIR). *Neurocomputing*. **452** pp. 675-689 (2021), https://www.sciencedirect.com/science/article/pii/S0925231220319044

[25] Radenović, F., Tolias, G. & Chum, O. CNN Image Retrieval Learns from BoW: Unsupervised Fine-Tuning with Hard Examples. *Computer Vision – ECCV 2016*. pp. 3-20 (2016)

[26] Kingma, D. & Ba, J. Adam: A method for stochastic optimization. *ArXiv Preprint ArXiv:1412.6980*. (2014)