Product Recognition for Unmanned Vending Machines

Chengxu Liu, Zongyang Da, Yuanzhi Liang, Yao Xue, Guoshuai Zhao, *Member, IEEE*, and Xueming Qian, *Member, IEEE*

***Abstract*— Recently, the emerging concept of "unmanned retail" has drawn more and more attention, and the unmanned retail based on the intelligent unmanned vending machines (UVMs) scene has great market demand. However, existing product recognition methods for intelligent UVMs cannot adapt to large-scale categories and have insufficient accuracy. In this article, we propose a method for large-scale categories product recognition based on intelligent UVMs. It can be divided into two parts: 1) first, we explore the similarities and differences between products through manifold learning, and then we build a hierarchical multigranularity label to constrain the learning of representation; and 2) second, we propose a hierarchical label object detection network, which mainly includes coarse-to-fine refine module (C2FRM) and multiple granularity hierarchical loss (MGHL), which are used to assist in capturing multigranularity features. The highlights of our method are mine potential similarity between large-scale category products and optimization through hierarchical multigranularity labels. Besides, we collected a large-scale product recognition dataset GOODS-85 based on the actual UVMs scenario. Experimental results and analysis demonstrate the effectiveness of the proposed product recognition methods.**

***Index Terms*— Large-scale product recognition, multiple granularity, object detection.**

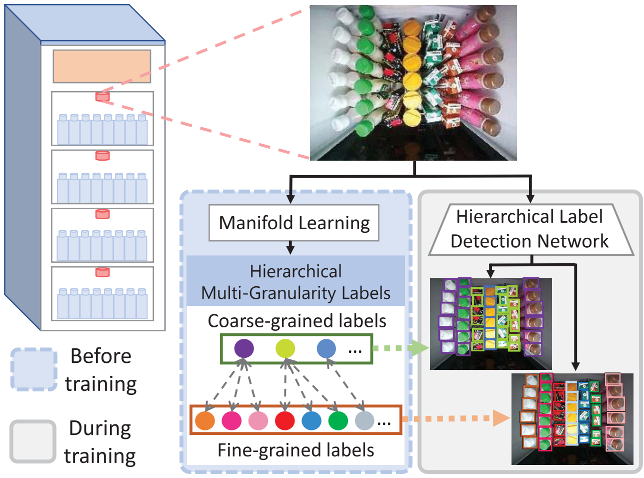


Fig. 1. Brief overview of our method. Before training, manifold learning is used to build hierarchical multigranularity labels. During training, the hierarchical label detection network is used to learn the location and classification of products.

I. Introduction

WITH the rapid development of computer vision and digital image processing based on deep learning in recent years. Technologies about product recognition related to intelligent unmanned vending machines (UVMs) are rapidly emerging. The intelligent UVMs based on computer vision have been successfully commercialized in some places and brought more and more convenience to users. However, for traditional UVMs, the process usually relies on mechanical tools and a lot of sensors, and it has the disadvantage of purchasing one item at a time, and the products that can be sold are fixed. Different from traditional UVMs, the core technology of unmanned retail based on intelligent UVMs scene is to recognize the products in the image collected by the camera [1], [2], and have the following advantages: 1) it combined with deep learning have the superiority of interaction and selectivity for the customer; 2) in addition, it can monitor the number of products in real-time, and efficiently customize the supplement plan, saving a lot of costs; and 3) it can boost the potential commercial applications by data of customer purchase behavior [3]–[5]. Therefore, proper object detection and recognition method is the key to realizing intelligent UVMs settlement. In this work, we focus on the large-scale categories of product recognition based on the intelligent UVMs scenario by combining detection network and manifold learning as shown in Fig. 1.

There are many studies on intelligent UVMs. For the existing product recognition methods, they mostly focus on smart unstaffed retail shop [6]–[8], or consider the customer's

Manuscript received 11 January 2021; revised 5 July 2021 and 3 January 2022; accepted 8 June 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2018AAA0101501; in part by NSFC under Grant 61772407, Grant 61732008, and Grant 62103317; in part by the Science and Technology Program of Xi'an, China, under Grant 21RGZN0017; in part by the Pazhou Laboratory, Guangzhou; and in part by the Microsoft Research. *(Corresponding author: Xueming Qian.)*

Chengxu Liu, Zongyang Da, and Yao Xue are with the School of Information and Communication Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: liuchx97@gmail.com; dzy1134483011@stu.xjtu.edu.cn; xueyao@xjtu.edu.cn).

Yuanzhi Liang and Guoshuai Zhao are with the School of Software Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: liangyzh13@stu.xjtu.edu.cn; guoshuai.zhao@xjtu.edu.cn).

Xueming Qian is with the Ministry of Education Key Laboratory of Intelligent Networks and Network Security, School of Information and Communication Engineering, Xi'an 710049, China, and also with the SMILES Laboratory, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: qianxm@mail.xjtu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2022.3184075.

Digital Object Identifier 10.1109/TNNLS.2022.3184075

purchasing process [1], [9]. Their tasks were used to recognize products that contain only ten distinct categories, which made it impossible to achieve good results in large-scale category recognition, especially in the case of dense placement. For existing product datasets, they either focus on the side of the products [10]–[12] or the categories are few and sparsely distributed [1]. Especially in the scenario of intelligent UVMs, Zhang *et al.* [1] constructed a dataset for multiclass beverage detection. The datasets comprise ten categories of beverages in the Chinese market, with an average of 4.56 instances per image. Although the number of images has more than 30k, the product categories are very few and sparsely placed. For practical applications, this is not appropriate, because businesses always want to place more products and support more abundant products. Our dataset can achieve nearly a hundred kinds of large-scale product recognition and cover a total of 85 categories of products. They include mineral water, beverages, chewing gum, and milk. The products are densely laid out, with an average of 22.97 instances per image, this is in line with the actual needs. Therefore, the existing works based on intelligent UVMs have the following problems.

1) The existing works support limited product categories. This narrows the range of products that customers can choose.
2) The existing methods do not consider the similarity and differences between multigranularity features of products. Generally, it is difficult to learn different fine-grained features for similar products. Therefore, the high similarity between different classes leads to poor performance (such as beverages of the same brand with different flavors).

The main challenge of this work is that the high intraclass variance due to the angle and position, and the low interclass variance due to the appearance, especially for the high variety of products. Therefore, considering these multigranularity features of product, some factors should be noticed.

1) It is difficult to learn the fine-grained differences between product categories, especially for the products from the same class or the same brand. For example, as shown in Fig. 2, all bottled water has the same top contour structure, and most of their bottle caps are white in color. Master kong jasmine honey tea and jasmine tea have the same bottle cap and similar drink colors.
2) The potential coarse-grained correlation should be considered. For products that are very similar in appearance, often have the same coarse-grained characteristics and tiny fine-grained differences, both of which coexist. For example, the top contour structure of mineral water is round, but the colors of different categories have unique fine-grained features. There are both commonness and differences among them, so a proper approach to establish the constraint relationship is crucial.

In general, focusing on the shortcomings of existing methods and challenges, we have made efforts in the following aspects.

1) Inspired by t-distributed stochastic neighbor embedding (t-SNE) [13] in manifold learning, we exploit the multigranularity characteristics of products and propose a scheme that explores similarities and differences between products and build a hierarchical multigranularity label. The main reason for using t-SNE is that it can learn the distribution of data in the low-dimensional manifold space through nonlinear dimensionality reduction and retain the essential characteristics of data. It can be used to mine feature similarity among product data and generate hierarchical multigranularity labels to optimize the network's learning of product features.
2) For making full use of the multigranularity features, we propose a hierarchical label detection network.

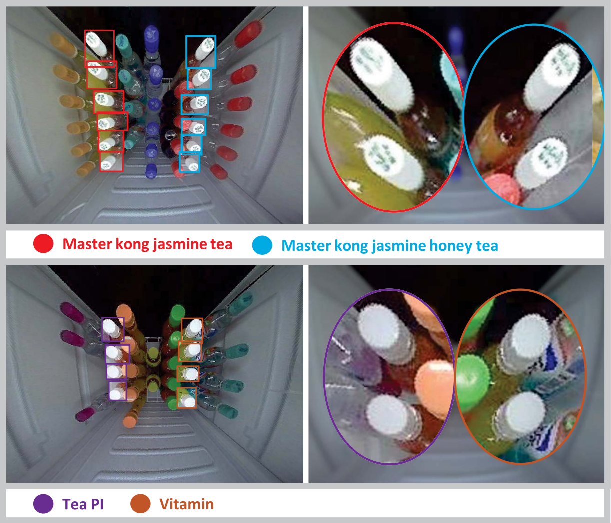


Fig. 2. Illustration of the similar products, in which the left figure of each row indicates the position and category of products, and the right side is the enlarged images.

On the whole, the highlight of our method is that it considers the potential similarity between large-scale category products and optimizes the learning through hierarchical multigranularity labels. In addition, a hierarchical label detection network is proposed, the potential multigranularity representation constraint information is added to refine the features.

In more detail, our method can be divided into two parts.

1) A scheme is used to generate hierarchical multigranularity labels. It first explores the high-level differences of products and maps them to the low-dimensional space through manifold learning. Then combine some products with similar feature distribution and generate coarse-grained labels. Finally, combine with the original annotations of the products themselves, we generate the hierarchical multigranularity label for each item. It contains multigranularity representations and constraints of the product, and it will be used as annotation information to guide the training.
2) A hierarchical label object detection network is used to introduce the multigranularity annotation information of products in the training stage and mainly includes C2FRM and MGHL.

The C2FRM is designed to output multigrained categories, and optimize the multigrained learning from coarse to fine during training. The MGHL is designed to constrain the hierarchical interrelationship between multigrained labels. Besides, to prove the effectiveness of our method, we conducted experiments on the GOODS-85 dataset, which we collected based on the actual UVMs scenario.

Our main contributions are as follows.

1) We explore the high-level differences and the potential similarity between the products, build a hierarchical multigranularity label inspired by manifold learning. Optimize the learning of multigranularity features of products.
2) We propose a hierarchical label detection network, which mainly includes coarse-to-fine refine module (C2FRM) and multiple granularity hierarchical loss (MGHL). They, respectively, optimize the network's learning of multigrained features of products and consider the hierarchical constraints interrelationship between multigranularity labels.
3) Extensive experiments demonstrate the effectiveness of the proposed method on the GOODS-85, which we collected based on the actual UVMs scenario and includes a total of 85 products. Experimental results show that our model obtains better performance than existing methods.

The rest of the article is organized as follows. Related work is reviewed in Section II. The proposed method is elaborated in Section III. The dataset collected based on the actual UVMs scenario is elaborated in Section IV. Experimental evaluation, analysis, and the discussion of the related parameters are presented in Section V. Finally, we conclude this work in Section VI.

II. Related Work

In this article, we first consider the characteristics of product images in intelligent UVMs and get the multigranularity representation of products inspired by manifold learning. Additionally, we propose a product detection network, which optimized the network for the learning of products' multigranularity features. Thus, in this section, we mainly introduce the related work on product recognition based on intelligent UVMs and object detection. Additionally then, we give a brief overview of multigranularity representation and manifold learning method and their application in various fields.

A. Product Recognition Based on Intelligent UVMs

The core technology of unmanned retail based on intelligent UVMs scene is to recognize the products in the image collected by the camera. Aiming at the intelligent UVMs in the unmanned retail industry, many works have made many contributions, which mainly include two parts: 1) the product datasets and 2) the existing product recognition method.

For existing products datasets, Goldman *et al.* [10] assembled a dataset and benchmark containing images of supermarket shelves. It contains a total of 110712 categories of products, with an average of 147.2 instances per image. Wei *et al.* [11] proposed a new dataset, which includes 200 categories of products for the automatic checkout task. Unlike our work, it has the features of the products in multiple perspectives, not only the top contour structure information. Zhang *et al.* [1] considered the real-world scenarios of UVMs, and constructed a large-scale dataset for multiclass beverage detection. The datasets comprise ten categories of beverages in the market of China, with an average of 4.56 instances per image. Different from our work, we collect a dataset covering a total of 85 categories of products. They include mineral water, beverages, chewing gum, and milk. The products are densely laid out, with an average of 22.97 instances per image.

Many works have made many contributions to product recognition. Aiming at exploring the feasibility of implementing the unstaffed retail shopping style, Liu *et al.* [6] proposed a smart unstaffed retail shop scheme. Li *et al.* [7] proposed a new data priming method to solve the domain adaptation problem in products' automatic checkout. Besides, Zhang *et al.* [1] divided the related tasks of customers in the purchase process into static detection and dynamic classification. Kim *et al.* [9] proposed a system to recognize purchasing behavior by detecting and tracking products in real-time using only camera sensors. Li *et al.* [14] proposed a backbone network of DrtNet, which adopts deformable convolution and group normalization layers for detecting beverages. Liu *et al.* [15] proposed a binocular camera system to solve the problems of distortion and coverage caused by the monocular camera in product recognition. Unlike the existing method, our work focuses on proposing a method for large-scale categories of product recognition based on Intelligent UVMs, and effectively improves the recognition performance. Use only information from one camera for static product detection once during the purchase process.

B. Object Detection

The key technology of product recognition based on intelligent UVMs scene is object detection. Recently, a series of object detection methods emerge in an endless stream and are widely used in the industrial field. Among them, the one-stage object detection methods based on anchor mechanism, such as SSD [16], DSSD [17], YOLO [18]–[20], RetinaNet [21], etc., not only has good detection accuracy, but also greatly improves the speed of object detection. In addition, the two-stage object detection methods based on anchor mechanism, such as Faster R-CNN [22], FPN [23], etc., have always occupied the highest results of general object detection. Then with the advent of CornerNet [24], object detection entered the era based on anchor-free, and more advanced detection methods CenterNet [25], ExtremeNet [26], FCOS [27] achieved better results. These object detection methods have their advantages, which gradually promote the development of computer vision.

More importantly, it is also very important in the industrial field. Hu *et al.* [28] provided a survey, which exploits deep learning for cancer detection and diagnosis. Said and Barr [29] applied deep learning to pedestrian detection. Based on the real-world surveillance video, Mabrouk and Zagrouba [30] conducted abnormal behavior recognition through the intelligent video surveillance system. Zhang *et al.* [1] and Kim *et al.* [9] applied object detection method in the computer vision to UVMs. However, they did not fully consider the large-scale categories of products in the container. Combined with the multigranularity features of products, we propose a better hierarchical label detection network for large-scale categories products.

C. Manifold Learning

In order to generate multigranularity representations of products, manifold learning is used to explore similarities and differences between products using high-level features.

Manifold learning assumes that the data of interest actually lie on an embedded nonlinear manifold within the higher-dimensional space. Its main application is to learn the distribution of data in the low-dimensional manifold space through nonlinear dimensionality reduction and to retain the essential characteristics of data. It has been applied in various fields. Yang *et al.* [31] proposed a semi-supervised algorithm called ranking with local regression and global alignment (LRGA) to learn the manifold space for data ranking. Hou *et al.* [32] first attempted to explore the manifold in the label space in multilabel learning. Zhao *et al.* [33] applied manifold learning to transfer learning and reduced the distribution difference between the source domain and target domain. He *et al.* [34] proposed a PolSAR image classification method combining nonlinear manifold learning with a fully convolutional network. It is clear that manifold learning can well reflect the essential characteristics of high-dimensional feature data and has high applicability.

Among the manifold learning methods, t-SNE [13] is a highly feasible and scientific way of nonlinear dimension reduction and visualization. Pezzotti *et al.* [35] presented a novel approach to the minimization of the t-SNE objective function that has linear computational complexity. Priam [36] believed that t-SNE and its variants lead to competitive nonlinear embeddings which were able to reveal the natural classes. Li and Yan [37] proposed a method for 3-D shapes isometric deformation using t-SNE based on inner distance (In-tSNE). In this work, we use t-SNE to mine feature similarity among product data and generate hierarchical multigranularity labels to optimize the network's learning of product features.

D. Multigranularity Representation

Multigranularity representation is a kind of method combining multigranularity features to study and analyze data. In this work, in order to obtain better features, we explore the multigranularity representation through manifold learning.

The multigranularity features can effectively improve the learning degree of the network, which can be applied to other fields. Wehrmann *et al.* [38] proposed architecture for hierarchical multilabel classification and discovered local hierarchical class relationships and global information. Yu *et al.* [39] propose the spatial pyramid structure to enhance the vector of locally aggregated descriptors (VLADs) for place recognition. Wang *et al.* [40] and Li *et al.* [41] proposed the feature learning strategy integrating discriminative information with various granularities. Yang *et al.* [42] introduced multiple granularity analysis frameworks for video segmentation in a coarse-to-fine manner. Lue *et al.* [43] proposed a strategy integrating global and local information in different granularities and spatial constraints for clothes retrieval. Wang *et al.* [44] constructed multigranularity descriptors by mining the subordinate level labels for fine-grand classification. Yu *et al.* [45] devise a hierarchical deep word embedding (HDWE) model which is a coarse-to-fine predictor to address click feature prediction for fine-grand classification.

These existing methods fail to consider the potential constraints between granularity features of different levels. Most importantly, its approach is not well extended to detection tasks. In our work, we first mined the multigranularity features of the data and then used it to optimize the training of our proposed hierarchical label detection network.

III. Method

A. Overview

Our method consists of two major parts: hierarchical multigranularity labels and a hierarchical label detection network.

1) In the part of hierarchical multigranularity labels, we propose a scheme for generating hierarchical multigranularity labels. It first explores the high-level differences of products and maps them to the low-dimensional space through manifold learning, and then combines some products with similar distribution and generates coarse-grained labels. After being combined with the original annotations of the products themselves, each product contains coarse-grained and fine-grained categories labels, and these two labels have a subordinate relationship. Then we combine these two kinds of labels according to their affiliation, and finally, generate a hierarchical multigranularity label for each item. It is not only the representation of products in multiple granularities but also has the corresponding relation between different granularities. It will be used as annotation information to guide the training.
2) Hierarchical label detection network introduces the multigranularity annotation information of products in the training stage, which mainly includes C2FRM and MGHL. As shown in Fig. 3. The C2FRM outputs coarse-grained and fine-grained categories, and at the same time optimize the network's learning of multigrained features of products from coarse to fine. The MGHL is designed to consider the hierarchical constraints interrelationship between coarse-grained and fine-grained labels, and further optimize the learning of hierarchical multigranularity features. Eventually, the total loss function used in training is described in detail.

B. Hierarchical Multigranularity Labels

In the real UVMs scene, products have potential similarities, so it is necessary to consider their hierarchical multigranularity representation. In this section, we propose a scheme that is used to generate hierarchical multigranularity labels. It mainly consists of three parts: high-dimensional feature extraction, feature reduction, and feature clustering: 1) high-dimensional feature extraction firstly cuts out the product area in the image and then extracts the features of the image containing only a single product through CNN; 2) inspired by t-SNE [13] in manifold learning, feature reduction reduces the high-dimensional features extracted in the previous step to 2-D space and visualizes them; and 3) hierarchical label division is used to cluster 2-D features after dimension reduction, and the result of dividing is the coarse-grained category label corresponding to the products. Finally, combining coarse-grained and fine-grained labels, we get hierarchical multigranularity labels as the final annotations. The detailed methods are as follows.

1) High-Dimensional Feature Extraction: To effectively extract the features of each product, we crop images through the bounding box from annotations and get an image gallery containing only one product per image. Then we randomly select 50 images of each product from the image gallery as the collection of images of this product. So we end up with a set of images of all the products.

In our work, ResNet [46] pre-trained on ImageNet is used as the feature extraction network, and each image can be represented as a higher-dimensional vector. It is a mapping from image to geometric space and can be represent as $\mathscr{F}(\cdot)$. In detail, Img_n^k represents the nth image in the set of product category k, where the value of $n \in \{1, 2, \ldots, 49, 50\}$, and the value of $k \in \{1, 2, \ldots, K-1, K\}$, where K is the total quantity of product category. The output feature $\boldsymbol{x}_i$ represents nth image feature in the set of product category k, it can be represented as follows:

$$\boldsymbol{x}_i = \mathscr{F}(\mathrm{Img}_n^k) \tag{1}$$

where $\boldsymbol{x}_i \in R^d$, and the d is the dimension of the feature, $i \in \{1, 2, \ldots, 50, 51, \ldots, 50 \times K\}$ is equal to $50 \times (k-1) + n$. It is worth noting that the dimension of this feature is high, and it is difficult to mine the potential similarity between products by high dimensional features.

2) Feature Reduction: Feature reduction is used to better mine and represent the potential feature similarity between high-dimensional features, inspired by t-SNE [13] in manifold learning. It first constructs the distribution of the data in high and low dimensional spaces, respectively, and then aims to fit the two distributions to the maximum extent possible. It is a highly feasible and scientific way of nonlinear dimension reduction and visualization.

In the high-dimensional space, the higher dimensional feature $\boldsymbol{x}_i$ that we got in the previous step forms a feature set that can be represented as $\mathbb{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{50\times K}\}$. For two elements $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in $\mathbb{X}$, we use $\boldsymbol{p}_{ij}$ to represent the distribution probability between them. The purpose of this design is to satisfy the symmetry between the difference, and $\boldsymbol{p}_{ij}$ can be represented as follows:

$$\boldsymbol{p}_{ij} = \frac{\boldsymbol{p}_{j|i} + \boldsymbol{p}_{i|j}}{2} \tag{2}$$

where the conditional probability $\boldsymbol{p}_{j|i}$ to represent the probability that when $\boldsymbol{x}_i$ is centered, $\boldsymbol{x}_j$ is chosen as its neighbor. $\boldsymbol{p}_{i|j}$ is exactly opposite with $\boldsymbol{p}_{j|i}$. The difference of product features can be modeled as Gaussian distribution. Mathematically, take $\boldsymbol{p}_{j|i}$ for example, it can be represented as follows:

$$\boldsymbol{p}_{j|i} = \frac{\exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2/2\boldsymbol{\sigma}_i^2)}{\sum_{m\neq i}\exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_m\|^2/2\boldsymbol{\sigma}_i^2)} \tag{3}$$

where $\boldsymbol{\sigma}_i$ is the variance of the Gaussian when $\boldsymbol{x}_i$ is centered.

In the low-dimensional space, we assume that the feature set after dimension reduction is represented as $\mathbb{Y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_{50\times K}\}$, and $\boldsymbol{q}_{ij}$ represents the distribution probability of $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$. Considering the crowding problem [13] of data during the dimensional transformation, the difference between the feature is modeled as a t-distribution with one degree of freedom, and $\boldsymbol{q}_{ij}$ can be represented as follows:

$$\boldsymbol{q}_{ij} = \frac{(1 + \|\boldsymbol{y}_i - \boldsymbol{y}_j\|^2)^{-1}}{\sum_{m\neq l}(1 + \|\boldsymbol{y}_m - \boldsymbol{y}_l\|^2)^{-1}} \tag{4}$$

where the $\boldsymbol{y}_i \in R^2$, $\boldsymbol{y}_j \in R^2$, $\boldsymbol{y}_m \in R^2$ and $\boldsymbol{y}_l \in R^2$ are the elements of $\mathbb{Y}$.

Our aim is to simulate the data distribution of high dimensional space in low dimensional space and explore the difference between different products. Therefore, Kullback-Leibler Divergence can effectively fit the two distributions, and the specific cost function Cost can be represented as follows:

$$\mathrm{Cost}(\boldsymbol{p}\|\boldsymbol{q}) = \sum_i \sum_j \boldsymbol{p}_{ij} \log \frac{\boldsymbol{p}_{ij}}{\boldsymbol{q}_{ij}} \tag{5}$$

by minimizing the $\mathrm{Cost}(\boldsymbol{p}\|\boldsymbol{q})$, the optimal feature set $\mathbb{Y}$ is the feature set after dimension reduction. The visualization of data distribution after feature reduction is shown in Fig. 4(a), in which the distribution of 85 products in total.

3) Hierarchical Label Division: To divide the hierarchy labels reasonably, we visualized the distribution of the product after feature reduction, as shown in Fig. 4(a). Among them, different categories are represented by different numbers. Some products are usually distributed adjacent and have a potential correlation. Therefore, based on our experience, we select 10, 8, 6, 4, 2 coarse-grained categories, respectively, and use k-means clustering to divide the products. The results with hierarchical label division is shown in Fig. 4(b)–(f). Among them, different colors indicate that products are divided into different categories. After partitioning, based on feature differences, we generate multigranularity labels with hierarchical relationships. It includes the multigranularity labels and the constraints between different hierarchies.

C. Hierarchical Label Detection Network

In this section, we propose a hierarchical label detection network for product recognition, and at the same time, use hierarchical multigranularity labels obtained from the above section. As shown in Fig. 3. The base network is VGG16. RPN and ROI Align are the same as Faster R-CNN [22] and Mask R-CNN [47].

During inference, the hierarchical label detection network extracts features through the base network, and the RPN is used for regional proposal, and then the C2FRM is input through the ROI Align. C2FRM will output probability with different granularity, and we associate them as the final product score. During training, in addition to the MGHL, regression loss and classification loss were also used. MGHL is used to constrain the multigranularity score output of C2FRM,

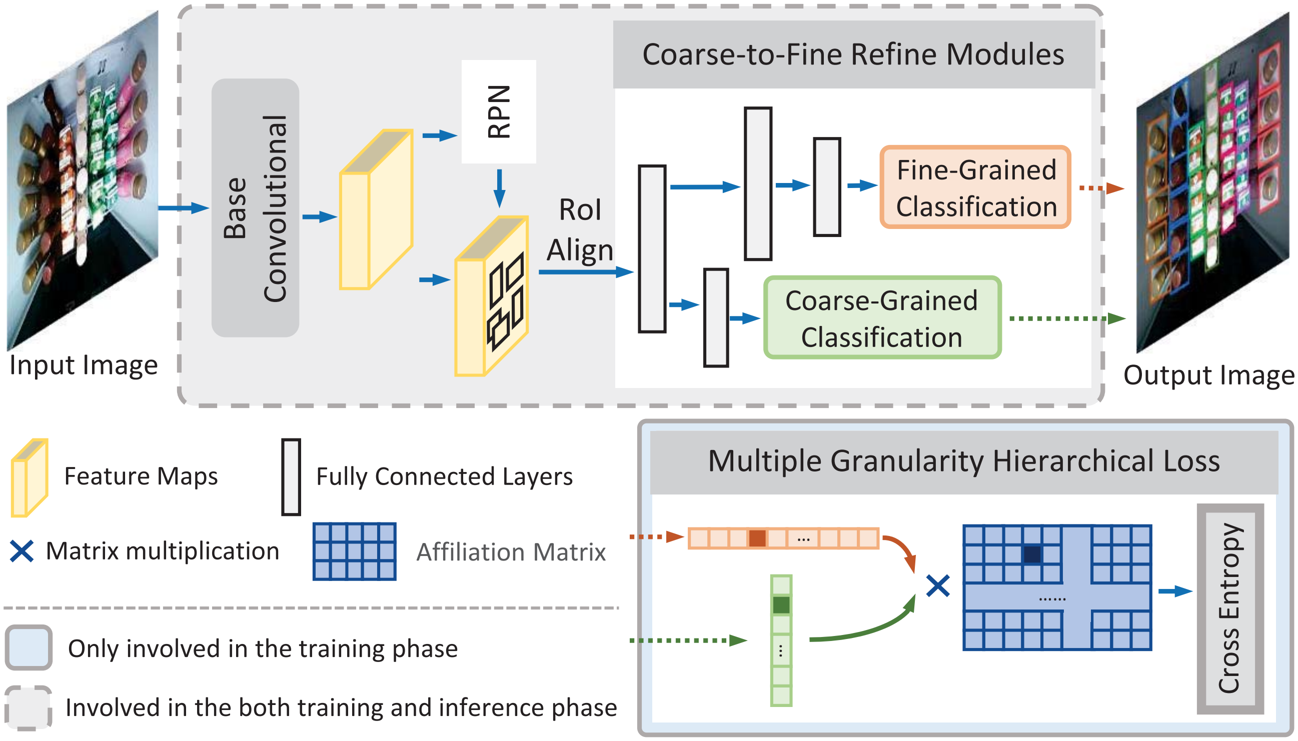

Input Image
Base Convolutional
RPN
RoI Align
Coarse-to-Fine Refine Modules
Fine-Grained Classification
Coarse-Grained Classification
Output Image
Feature Maps
Fully Connected Layers
Matrix multiplication
Affiliation Matrix
Only involved in the training phase
Involved in the both training and inference phase
Multiple Granularity Hierarchical Loss
Cross Entropy

Fig. 3. Overview of our hierarchical label detection network, which mainly includes C2FRM and MGHL. In the process of training and reasoning, C2FRM is used to output the multigranularity labels of each instance. MGHL only serves to guide learning during training.

specifically, the matrix product is used to establish the affiliation matrix and calculate its cross-entropy. In this way, the constraint relationship between the multigranularity features is further considered.

1) Coarse-to-Fine Refine Module: The C2FRM is used to output the category labels with different granularity. Its structure consists of fully connected layers, and its dense connection mode makes it have an excellent nonlinear fitting ability. We output the hierarchical multigranularity labels of the products in an asymptotic manner, with the purpose of making C2FRM guide the learning of features in a coarse-to-fine way, which is equivalent to adding additional product information.

As shown in Fig. 3, the whole C2FRM is composed of four fully connected layers. Among them, two layers are used to output coarse-grained classification and three layers are used to output fine-grained classification, and they share the parameters of the first layer. Hierarchical multigranularity labels are used to constrain the multigranularity score output of C2FRM. During training, this part includes three parts of the loss function, the classification cross-entropy of coarse granularity and fine granularity, respectively, and the MGHL. The training process is the process of refinement of better features.

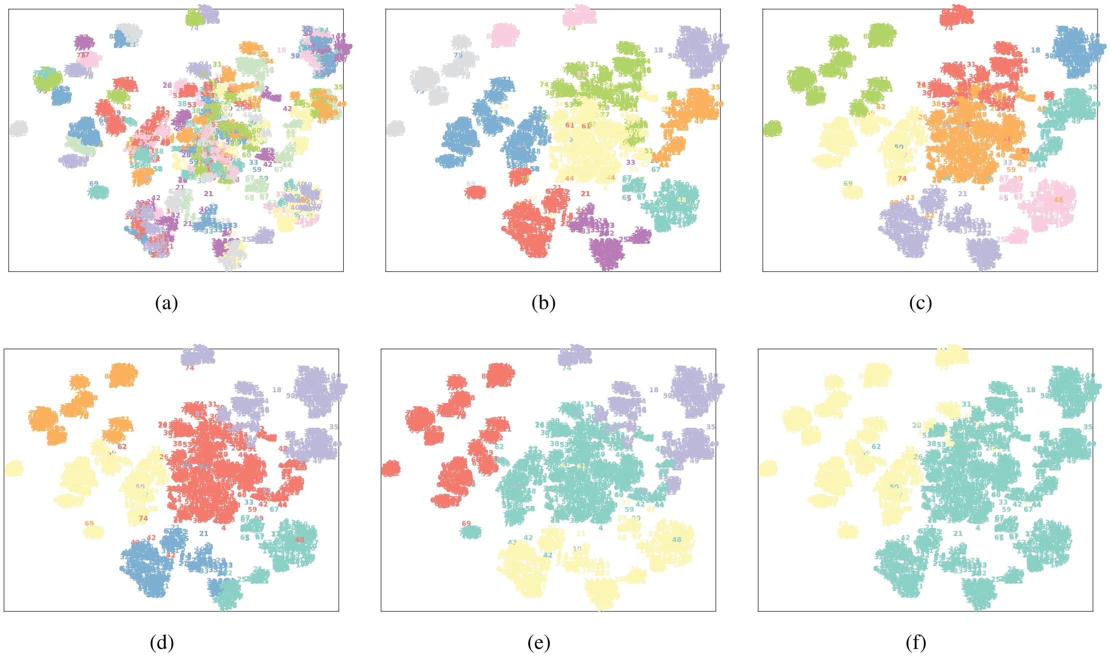
2) Multiple Granularity Hierarchical Loss: We explore the potential relationship between different granularity categories and propose multigranularity hierarchical loss. It first calculates the affiliation matrix of category scores under different granularity by matrix product, and then calculates the cross-entropy of the affiliation matrix as the final result. We discuss the impact of the different losses in our method in Section V-C5.

In detail, for a given image, let $\boldsymbol{p}_f \in R^{K_f \times 1}$ represents the fine-grained categories score of the output, $\boldsymbol{g}_f \in R^{K_f \times 1}$ represents the ground truth of fine-grained categories label. And $\boldsymbol{p}_c \in R^{K_c \times 1}$ represents the coarse-grained categories score of the output, $\boldsymbol{g}_c \in R^{K_c \times 1}$ represents the ground truth of coarse-grained categories label. Among them, K_f and K_c are the number of fine-grained and coarse-grained categories, respectively. The MGHL can be represented by $\mathrm{MGHL}(\boldsymbol{p}_c, \boldsymbol{p}_f)$ as follows:

$$\mathrm{MGHL}(\boldsymbol{p}_c, \boldsymbol{p}_f) = -\sum (\boldsymbol{g}_c \cdot \boldsymbol{g}_f^T) \log(\boldsymbol{p}_c \cdot \boldsymbol{p}_f^T) \tag{6}$$

where the $\boldsymbol{g}_c \cdot \boldsymbol{g}_f^T \in R^{K_c \times K_f}$, $\boldsymbol{f}_c \cdot \boldsymbol{f}_f^T \in R^{K_c \times K_f}$ are the ground truth of affiliation matrix and the output scores affiliation matrix, respectively. The affiliation matrix is a 2-D matrix with length K_c and width K_f. The MGHL is obtained by cross-entropy calculation on a 2-D affiliation matrix, which is equivalent to increasing the length of the label compared to the cross-entropy calculation based on 1-D labels.

The affiliation matrix represents the dependency relationship between the coarse-grained and fine-grained category, with a value of 1 if and only if the product belongs to both right coarse-grained and fine-grained classes, and 0 in the remaining cases. This shows that $\mathrm{MGHL}(\boldsymbol{p}_c, \boldsymbol{p}_f)$ can effectively suppress the wrong dependency relationship of product categories under different granularity, and fully consider the potential affiliation between category labels under different granularity.


(a) (b) (c)

(d) (e) (f)

Fig. 4. Visualization of the distribution of instances categories in low dimensional space. (a) Shows the results of original product data in a low dimensional space, different categories are represented by different numbers. (b)–(f) Show the results with hierarchical label division when the coarse-grained category is 10, 8, 6, 4, 2. Among them, different colors indicate that products are divided into different categories.

3) Overall Loss function: We introduce in detail the overall loss function of our proposed hierarchical label detection network during training. The overall loss function consists of two parts: 1) one is the same loss on PRN as Faster R-CNN [22] and 2) the other part contains a total of four items. Specifically, in addition to the MGHL mentioned above, it also includes the bounding box regression loss, and the classification loss includes fine-grained and coarse-grained, respectively.

In detail, for a given image, the overall loss function can be represented by L_{all} as follows:

$$L_{\text{all}} = L_{\text{reg}}^{\text{RPN}} + L_{\text{cls}}^{\text{RPN}} + \text{MGHL} + L_{\text{reg}} + L_{\text{cls}}^{f} + L_{\text{cls}}^{c} \tag{7}$$

where the $L_{\text{reg}}^{\text{RPN}}$ and $L_{\text{cls}}^{\text{RPN}}$ are the regression and classification loss of the foreground on RPN as described in [22], the MGHL is described in detail above. The bounding box regression adopted smooth $L1$ loss function can be represented by L_{reg} as follows:

$$L_{\text{reg}} = \sum_{i \in \text{Positive}}^{N} \sum_{m \in \{cx, cy, w, h\}} \text{smooth}_{L1}\left(l_i^m - \widehat{g}_i^m\right) \tag{8}$$

where N is the number of matched positive boxes, the l and $\widehat{g}$ are the predicted box and the ground truth box, respectively, as the same as described in [22]. The box center (cx, cy), width w, and height h are the offsets used for regression. Besides, the L_{cls}^{f} and L_{cls}^{c} in (7) are the classification cross-entropy of fine-grained and coarse-grained, respectively. They can be unified and represented by L_{cls} as follows:

$$L_{\text{cls}} = -\sum_{j=1}^{C} g_j \log p_j \tag{9}$$

where C is the number of categories, g_j is the ground truth of category and p_j is the output scores. The training process is the process of refinement of better features.

IV. Dataset

In this section, to demonstrate the superiority of our method, we construct a dataset that includes large-scale categories of products. We briefly introduce the generation of our GOODS-85 dataset. It can be divided into three parts: the process of image collection on the actual container scenario, image correction to avoid overlapping bounding boxes, and the procedure of data annotation.

A. Image Collection

Our data acquisition platform is based on the container of a four-tier, with a 300 000-pixel high-definition fisheye camera mounted on the top and center of each floor of the container. The inner diameter of each floor is about 600 and 500 mm in length and width, and 350 mm in height. Similar to an actual unmanned intelligent container, the high-definition fisheye camera collects information directly above the products. Compare with ordinary high-definition cameras, high-definition fisheye cameras have a very wide field of view, which allows the container to carry more items in limited space, and can cover all products information in the container to a large extent.

We collect different images by changing the type and quantity of products in the container. An image is collected for each shift of the products placed. In the end, there were a total of 1047 pictures, each of a size of 640*480, covering a total of 85 items. They include mineral water, beverages,

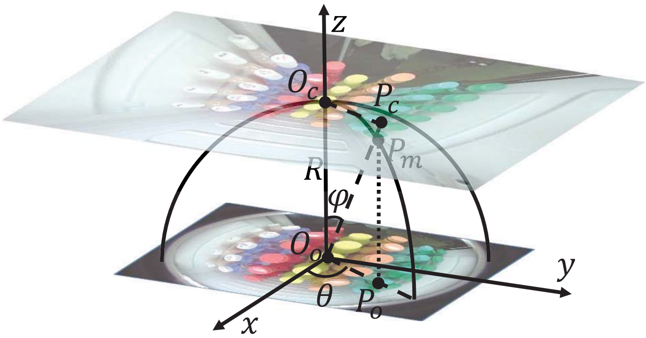


Fig. 5. Illustration of the spherical isometric projection correction model.

chewing gum, and milk. They are the most common products in the Chinese market.

B. Image Correction

The images collected by the high-definition fisheye camera have a very wide field of view, which can cover all the items in the container completely. However, it can be seen from the fisheye image that the products are staggered, and the visible area of the products at the boundary of the container is small, especially in the case of dense items. So when the bounding boxes are overlapped seriously, it will result in poor performance if only the original fisheye image is used.

In order to avoid this problem, inspired by [48], we adopt the idea of mathematical modeling to correct the distortion of the fisheye image. It can stretch the fisheye image from the center position to the surrounding position by spherical isometric projection correction model, so as to avoid the serious overlapping of bounding boxes caused by the dense spatial position, and improve the detection performance.

The spherical isometric projection correction is a nonlinear projection correction model, which describes the correspondence between the points on the spherical surface and the plane image. In this section, our aim is to map each point in the original image to the corrected plane. The model can be represented as Fig. 5, where O_o and P_o are the points on the original plane before correction, O_c and P_c are the points on the plane after correction, and P_m is the point on the sphere. R, φ, and θ are the distance and angle parameters of the sphere.

We set the original image in the xO_oy plane, the coordinate of P_o in the original image can be represented as (x_o, y_o). Then, the point P_o is mapped to point P_c after correction, and the coordinate of P_c in the corrected image can be represented as (x_c, y_c). The principle of spherical isometric mapping correction model can be explained as follows: for any point P_m on the surface of sphere, its deviation angle in the vertical direction can be represented as φ, and the relationship between φ and the radial distance O_oP_o in the image plane can be represented as follows:

$$\varphi = \frac{O_oP_o}{R} \times 90^\circ \tag{10}$$

where R is the radius of the view of the container in the original image.

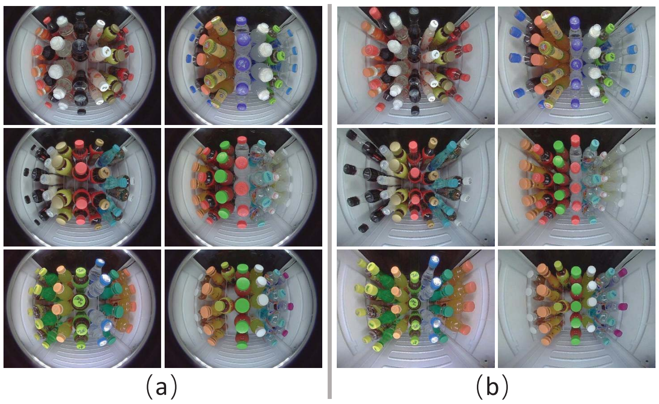


Fig. 6. Illustration of the sample images collected by us: (a) and (b) are examples of before and after the correction, respectively.

Then, the radial distance O_cP_c in the corrected image plane can be represented as follows:

$$O_cP_c = R \times \tan\varphi \tag{11}$$

then the horizontal and vertical coordinates x_c and y_c of point P_c in the corrected image can be calculated by the following equation, respectively:

$$x_c = O_cP_c \times \cos\theta \tag{12}$$

$$y_c = O_cP_c \times \sin\theta \tag{13}$$

where θ is the deflection angle of point P_o in the xO_oy plane. It can be represented as follows:

$$\theta = \tanh\frac{y_o}{x_o}. \tag{14}$$

Since the corrected image size tends to infinity theoretically, we only ensure that the products appear intact in the corrected area. After the above correction algorithm, point P_o in the original image is mapped to point P_c in the corrected image. Effectively avoids the problem of the small visual area of items placed on the boundary of the container and serious overlap of the bounding box, especially in the case of dense items. Some sample images are shown in Fig. 6.

C. Image Annotation Procedure

In the end, there are a total of 1047 pictures, each of a size of 640*480, covering a total of 85 items. They include mineral water, beverages, chewing gum, and milk. For each category of product in the image, we manually labeled the category to which it belongs. For properly placed items, we have labeled the top of the item area, and the bounding box can cover the top contour of the items well. For some items at the boundary of an image or obscured by others, we try to cover the area visible to the items as much as possible. More importantly, such placement rules are not allowed in the actual situation. Moreover, a total of 24 051 instances were labeled with category labels and bounding boxes.

The distribution of images and instances in the dataset is shown in Table I. There are about 100 to 1000 instances of each category. In addition, the number of products in each image is very dense, which can contain 57 products at most.

TABLE I

SAMPLE DISTRIBUTION OF IMAGES AND INSTANCES IN DATASET (IMAGES INDICATE THE NUMBER OF IMAGES, AND OBJECT DENOTES THE NUMBER OF INSTANCES)

Data	Type	Distribution		Category	Instances per image
		trainval	test		
Detection	Image	733	314	85	22.97
	Object	17295	6756		

TABLE II

QUANTITATIVE RESULTS IN TERMS OF MAP IN COMPARISON OF STATE-OF-THE-ART OBJECT DETECTION MODELS ON GOODS-85 DATASET

Method		Backbone	Size	mAP
Two-stage	Faster R-CNN [22]	VGG16	600	91.9%
One-stage	SSD [16]	VGG16	512	91.2%
	YOLOv3 [20]	Darknet-53	608	89.1%
	RetinaNet [21]	ResNet50-FPN	600	84.9%
Anchor-free	CornerNet [24]	Hourglass-104	511	92.1%
	CenterNet [25]	Hourglass-104	511	90.4%
	FCOS [27]	ResNet50-FPN	600	90.4%
Ours		VGG16	600	**93.7%**

V. EXPERIMENTS

In this section, we first briefly introduce our experimental settings. And then, in the results and analysis section, we compare other methods and ablation experiments. Finally, we discuss the relative parameters in our method.

A. Experimental Settings

1) Compared Methods: A brief introduction of the compared approaches is as follows.

1) *Faster R-CNN [22]:* Two-stage object detection network, which introduces a region proposal network (RPN) that shares convolutional features and enables nearly cost-free region proposals.
2) *SSD [16]:* One-stage object detection network, which combines predictions from multiple feature maps with different resolutions to naturally handle objects of various sizes.
3) *YOLOv3 [20]:* One-stage object detection network, combined with multiscale predictions and a better backbone classifier, is extremely fast and accurate.
4) *RetinaNet [21]:* One-stage object detection network, adopt the feature pyramid network (FPN) from [23] as the backbone network and efficiently constructs a rich, multiscale feature pyramid from a single resolution input image.
5) *CornerNet [24]:* Approach to object detection based on anchor-free, which detects an object bounding box as a pair of key points, the top-left corner, and the bottom-right corner, using a single convolution neural network.
6) *CenterNet [25]:* Approach to object detection based on anchor-free, which explores the central part of a proposal, use a triplet, instead of a pair, of key points to represent each object.
7) *FCOS [27]:* Approach to object detection based on anchor-free, which detects an object bounding box by predicting the deviation of a pixel to the center of its corresponding bounding box.
8) *Ours:* Our method for hierarchical label detection is based on the actual UVMs scene. Based on the visual characteristics of products, we use manifold learning to generate hierarchical multigranularity labels, propose C2FRM and MGHL to optimize the learning of product features during training.

2) Performance Evaluation: For performance evaluation, a widely used metric mean average precision (mAP) was calculated for products in our experiment. It can be calculated by the following formula:

$$\mathrm{mAP} = \frac{1}{m}\sum_{i=1}^{m} \mathrm{AP}_i \tag{15}$$

where m represents the number of categories of products. The idea of average precision (AP) can be conceptually regarded as calculating the area under the precision and recall curve of each product. The calculation formula of AP can be expressed as follows:

$$\mathrm{AP} = \frac{1}{11}\sum_{r\in\{0,0.1,\ldots,1\}} p_{\mathrm{interp}}(r) \tag{16}$$

where $p_{\mathrm{interp}}(r) = \max_{\widetilde{r}:\widetilde{r}\geq r} p(\widetilde{r})$ represent the maximum precision when recall equals to r. Among them, the precision and recall rate can be expressed as follows:

$$p = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{17}$$

$$r = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{18}$$

where TP, FP, and FN are the true positive, false positive, and false negative sample quantity of products, respectively.

3) Implementation Details: The proposed method is implemented by PyTorch. The base network adopts the pre-model of VGG16 in ImageNet. Each batch consists of one image on each GPU. We set the coarse-grained categories label to 6. We use the SGD optimization algorithm to train the network, and set the weight decay to be 0.0001 and momentum is set to be 0.9. For the detection head, the initial learning rate is 0.001 for the first 30 epoch, which decays by a factor of 10 for the next 20 and 10 epoch, and training stops after 60 epochs. All the experiments are conducted on a workstation with 8 GTX-1080Ti GPUs.

B. Results and Analysis

1) Objective Comparison: To prove the effectiveness of our method, we conducted experiments on the GOODS-85 dataset and SmartUVM [1] dataset. In Table II, we compare the classic object detection methods of one-stage, two-stage, and anchor-free in recent years. Compared with the classic two-stage, and one-stage methods for general object detection, such as Faster R-CNN [22], SSD [16], etc., the two-stage method has a better effect due to its strong adaptability to small-scale items. At the same time, the high similarity between categories

TABLE III
QUANTITATIVE RESULTS IN TERMS OF MAP IN COMPARISON OF STATE-OF-THE-ART OBJECT DETECTION MODELS ON SMARTUVM [1] DATASET

Method	Backbone	Size	mAP
Faster R-CNN [22]	VGG16	600	90.8%
SSD [16]	VGG16	512	91.2%
YOLOv3 [20]	Darknet-53	608	91.0%
RetinaNet [21]	ResNet50-FPN	600	90.9%
CornerNet [24]	Hourglass-104	511	91.3%
CenterNet [25]	Hourglass-104	511	91.4%
FCOS [27]	ResNet50-FPN	600	91.6%
Ours	VGG16	600	**92.1%**

and the anchor design also limits the effect of the one-stage methods. The method based on anchor free can also achieve good performance, such as CornerNet [24] reaching 92.1% of the mAP. However, when an image contains many of the same products, and they are placed densely, it will bring a certain difficulty to the selection of the center point and the matching of corner points. Based on the characteristics of the products, our method considers the similarity between products and achieves the best performance on the GOODS-85. Compared with the anchor Free method, our method improves by 1.6%. Compared with the one-stage and two-stage methods, our method improves by 1.8%–2.5%.

In Table III, we also compare typical detection methods. However, SmartUVM [1] dataset is different from ours, which only contains ten beverages with obvious differences, and its features are easier to learn. Also, since the dataset has fewer categories, the number of coarse-grained category labels in our method is set to 4. We can see that on SmartUVM [1], compared with Faster R-CNN [22], the one-stage based methods have better performance. In addition, The methods based on anchor free are still excellent. It can be found that although the advantages of our method cannot be fully exploited on this dataset, it can still bring about an improvement of about 0.5%.

2) Ablation Experiments: To prove the effectiveness of each part of the C2FRM and MGHL. In order to ensure that the experiment is not affected by other factors, we use the VGG16 as the backbone in these experiments. The experimental results are shown in Table IV. The first row indicates that the results of the experiment where the above mentioned are not used, and the product classification part uses only original category labels. The second row indicates that using the C2FRM as the output head of the product classification, and adding the hierarchical multigranularity labels of products can make the mAP improved by 1.0%. In addition, from the last row of the table, the addition of MGHL can increase the mAP by about 0.8% on this basis. Using two parts of C2FRM and MGHL at the same time can get better performance in product recognition.

Experiment results demonstrate that adding C2FRM can effectively improve the mAP of product recognition. Since forcibly learning the fine-grained difference between similar products will affect the stability of the network, and the C2FRM can use the hierarchical multigranularity labels, and its constraints improve the network stability. The MGHL is calculated based on the affiliation matrix, and it is used to constrain the affiliation between different granularities. The classification requirements are met only when the products are classified correctly under different granularities. In particular, comprehensive considerations of C2FRM and MGHL can bring a degree of improvement in this task.

TABLE IV
RESULTS OF ABLATION EXPERIMENTS USING C2FRM AND MGHL OF OUR METHOD ON GOODS-85 DATASET

Backbone(VGG16)	C2FRM	MGHL	mAP
✓			91.9%
✓	✓		92.9%
✓	✓	✓	**93.7%**

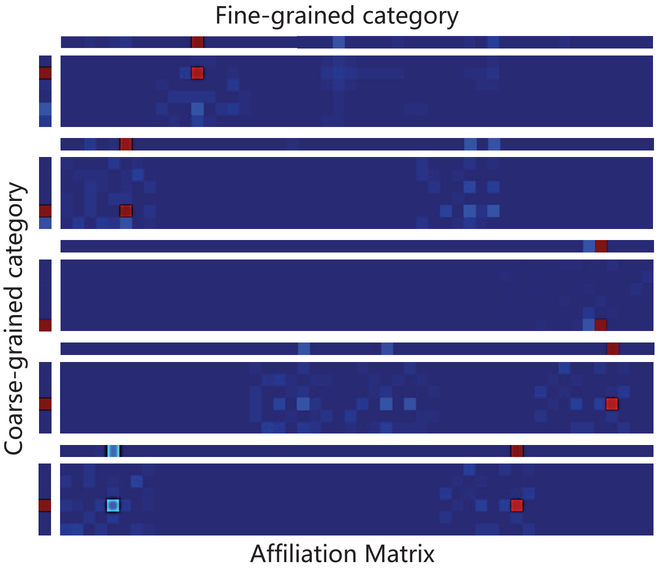


Fig. 7. Visualization of the hierarchical multigranularity outputs of products, which includes five products from top to bottom. The hierarchical multigranularity outputs of each product are composed of three parts. The left side is the output score of coarse-grained, the top side is the output score of fine-grained, and the rest is the affiliation matrix, which is obtained through the matrix product of the two kinds of output score.

C. Discussion

In this section, we completely discuss the works and contributions involved in our proposed methods. We split the discussion into six parts.

1) The effect of hierarchical multigranularity labels.
2) The heatmap visualization.
3) Discussion about our architecture of other backbones, mentioned in Section III-C. To illustrate the validity of our entire network structure, we have chosen another backbone as the base network to compare results.
4) Discussion of the impact of the hierarchical label division in our method, mentioned in Section III-B3.
5) Discuss the loss function of the affiliation matrix in the MGHL, mentioned in Section III-C2.
6) The effect of high-dimensional feature extraction network, mentioned in Section III-B1.

1) Effect of Hierarchical Multigranularity Labels: The first part of the discussion is a visual analysis of the effect of

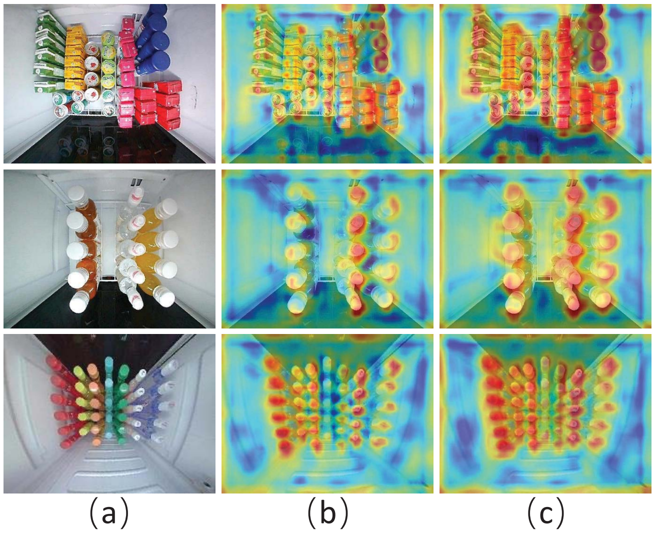


Fig. 8. Comparison of the feature map visualization: (a) is the original images, (b) is the result of not adding the C2FRM and MGHL, and (c) is the result of using the C2FRM and MGHL we proposed.

TABLE V

PERFORMANCE COMPARISON FOR DIFFERENT BACKBONE ON GOODS-85 DATASET. C2FRM AND MGHL ARE ABBREVIATIONS FOR C2FRM AND MGHL, RESPECTIVELY

Backbone	mAP					
	C2FRM	MGHL	C2FRM	MGHL	C2FRM	MGHL
			✓		✓	✓
VGG16	91.9%		92.9%		**93.7%**	
VGG16-FPN	92.2%		92.9%		**93.8%**	
VGG19	93.1%		94.0%		**94.2%**	
VGG19-FPN	92.8%		93.7%		**94.0%**	
ResNet50	80.6%		82.3%		**83.3%**	
ResNet50-FPN	91.8%		92.2%		**92.8%**	
ResNet101	83.0%		84.3%		**85.3%**	
ResNet101-FPN	92.2%		93.5%		**93.8%**	
ResNet152	84.1%		83.6%		**84.3%**	
ResNet152-FPN	92.1%		93.0%		**93.7%**	

hierarchical multigranularity labels. We have visualized the hierarchical multigranularity outputs, then artificially listed some cases of products, and analyzed the reasons why the hierarchical multigranularity labels are effective.

As shown in Fig. 7, we list the hierarchical multigranularity outputs of five different products in different images. The hierarchical multigranularity outputs of each product are composed of three parts. The left side is the output score of coarse-grained, the top side is the output score of fine-grained, and the rest is the affiliation matrix. Among the visualization region, the category score is higher, and the responses to the region are stronger. Similar products can be suppressed by the affiliation matrix, thus making the output more accurate. This is embodied that through hierarchical multigranularity outputs, the response of the correct category region is always high, while the response intensity of incorrect categories becomes weaker. Besides, It is worth noting that for similar products belonging to the same coarse-grained category, the stability of the model will be affected if the network forces them to learn, and the multiple constraints of hierarchical multigrained labels can effectively mitigate this effect.

2) *Visualization:* Different from the general object detection, the products are densely laid out and there is high intra-class variance and low inter-class variance. Therefore, accurate response to the location and features of the product is critical. Inspired by [49], whose validity has been proven. We calculated and visualized the heatmap of the feature map output by feature extraction, and the guidance of C2FRM and MGHL made the response of the product area stronger.

As shown in Fig. 8, we artificially selected some images including different product categories and different distributions for feature visualization. For each image, (b) is the heatmap without our C2FRM and MGHL, and (c) is the heatmap with our C2FRM and MGHL. Compared to not adding C2FRM and MGHL, the network pays more attention to the region getting more features of products. Besides, it can directly indicate the importance of the activation at spatial position effectively. The visualization results show that the areas of network attention are focused on products. It indicates that the C2FRM can play a role in making the network pay attention to products. Among the top region of products, feature extraction networks can extract features effectively, and there are almost no differences due to different product categories.

3) *Discussion About Other Backbones:* To illustrate the validity of our network structure and the universality of this method. In this section, we used different networks as the backbone for comparison. The experimental results compare our architecture with four different backbones, such as VGG19, ResNet50, ResNet101, and ResNet152. As shown in Table V, for each base network, the first column is the result using the convolutional neural network, which is used as the baseline. Our method result is represented as the last two columns in each backbone, and the C2FRM and MGHL as described in Section IV-C. Besides, in order to prove the validity of its structure, we did not add data augmentation in this experiment.

As the result shown in Table V, our method using VGG19 as the backbone has a better performance than other backbones. According to our analysis, the reason for the poor performance of using ResNet as a backbone is that ResNet is good at extracting deep semantic information of the object, and the products need more shallow features due to the small object size, dense placement, and high similarity between classes. At the same time, our structure can greatly improve the results of the original basic network. In addition, in order to prove this point of view, we added the additional FPN [23] structure. It can be seen that the FPN was used to integrate the shallow features, which brought a great improvement in ResNet. Compared with VGG19, VGG19-FPN has deeper network layers, thus increasing the difficulty of learning. It is worth noting that our method can be improved no matter what backbone is used. For our proposed C2FRM with the backbone of VGG19, the performance of mAP object detection results was increased by 0.9%, which proves the effectiveness of our proposed module can effectively refine the multigranularity

TABLE VI

PERFORMANCE COMPARISON OF USING DIFFERENT LOSS FUNCTIONS $f^{p}_{\text{Affm}}(\cdot)$ OF AFFILIATION MATRIX IN MGHL ON GOODS-85 DATASET

Method	mAP
Baceline(without MGHL)	92.9%
MGHL(Tversky Loss)	92.7%
MGHL(IoU Loss)	93.0%
MGHL(Focal Loss)	93.1%
MGHL(Dice Loss)	93.2%
MGHL(CE Loss)	**93.7%**

features of the products, it is effective in product detection and recognition. Consider the affiliation between multigranularity level labels. The model training by MGHL can add stronger constraint relationships to a certain extent and achieve better performance, and the performance of mAP is increased by 1.2% on the method.

4) Discussion About Hierarchical Label Division: As described above, the feature distribution of different products has some potential correlation. The distribution of the product after feature reduction is shown in Fig. 4(a). Therefore, we divide the coarse-grained categories of products according to their original fine-grained features. Most importantly, there are mutual constraints between coarse-grained and fine-grained labels, which will help the network learn the multi-grained features of the products.

The number of categories after the hierarchical label division is crucial, which determines whether the added coarse-grained constraint labels are appropriate. If the number of divisions is too small, it will introduce inappropriate over-constraint information. On the contrary, if the number of divisions is too large, it will not have sufficient effects. The appropriate hierarchical label division is obtained by the trade-off performance.

Fig. 9 shows the relationship between division selection and performance of product detection results. It is shown that regardless of whether MGHL is used, the division value of 6 can maximize the accuracy of products detection, and with the increase of it, the constraint information of multigranularity labels is getting weaker and weaker, and it is difficult to bring obvious improvement. Through the trade-off, we chose the division value as 6.

5) Discussion About Multiple Granularity Hierarchical Loss: As described in Section III-C2 (6), the MGHL first calculates the affiliation matrix of category scores under different granularity by matrix product and then calculates the cross-entropy of the affiliation matrix as the final result. In the MGHL, the choice of loss function of the affiliation matrix is an important and difficult issue. The final loss function can be represented by MGHL as follows:

$$\text{MGHL}(\boldsymbol{p}_c, \boldsymbol{p}_f) = f^{p}_{\text{Affm}}\left(\boldsymbol{p}_c \cdot \boldsymbol{p}_f^{T}\right) \tag{19}$$

where $f^{p}_{\text{Affm}}(\cdot)$ represent the loss function of affiliation matrix.

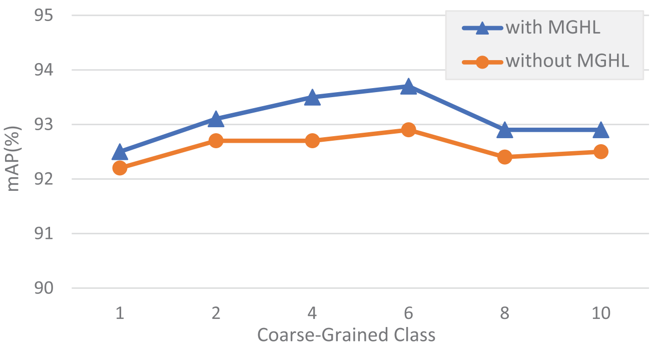


Fig. 9. Comparison of mAP of products detection using MGHL.

TABLE VII

PERFORMANCE COMPARISON OF USING DIFFERENT MODEL PRE-TRAINED ON IMAGENET IN HIGH-DIMENSIONAL FEATURE EXTRACTION

Feature Extracter	mAP	Feature Extracter	mAP	Feature Extracter	mAP
ResNet50	93.7%	DenseNet50	93.6%	ResNeXt50	93.0%
ResNet101	93.8%	DenseNet101	93.6%	ResNeXt101	94.0%
ResNet152	93.6%	DenseNet152	93.7%	ResNeXt152	93.8%

We compare several typical loss functions in the field of medical image segmentation to deal with sample imbalance such as Dice loss [50], Tversky loss [51] and the traditional classification methods CE Loss and Focal loss [21]. The performance is shown in Table VI. A proper loss function of the affiliation matrix is the main factor of product classification. The excessive and insufficient loss function of the affiliation matrix can lead to poor performance. Compared with the loss function used in the field of medical image processing, it is not suitable for this task because it mainly deals with the imbalance of labels and lacks the constraint ability for outputs. Focal loss increases the learning weights of positive and negative samples based on CE loss and it can force the network to learn the differences of samples. Therefore, the focal loss is not robust enough for outlier samples. The model training by cross-entropy can add stronger constraint relationships to a certain extent and achieve better performance, and the performance of mAP is increased by 0.5%–1.0% on the method.

6) Effect of High-Dimensional Feature Extraction Network: As described in Section III-B1, we use the model pre-trained on ImageNet to extract the high-dimensional features of the products, and each image can be represented as a higher-dimensional vector. What needs to be emphasized is that we consider the general consensus in various fields, and use ResNet50 as a tool to extract features, but it is not limited to ResNet50. To prove it, we compared networks with high level semantic feature extraction capabilities, such as ResNet [46], DenseNet [52] and ResNeXt [53]. As shown in Table VII, the experiment in this part retains the same experimental settings as above. It can be seen from the results that the use of different feature extraction networks to extract high-dimensional features has strong robustness and little impact on the final performance. This conclusion is also obvious, because it is only a preliminary feature extraction, and after the manifold learning, it will not even affect the division of the coarse-grained label of the item.

VI. Conclusion

In this article, firstly, we propose a scheme to mine the similarities and differences of products and generate more feasible information for guidance training. Secondly, we propose a hierarchical label detection network and optimize the network's learning of multigrained features of products. Finally, we collect a GOODS-85 dataset based on the actual UVMs scenario. Experimental results demonstrate that our method outperforms other state-of-the-art methods in two benchmarks. On GOODS-85, the use of our method get better performance of product recognition and improved the mAP to 93.7%. Compared with other typical detection methods, our methods were all improved the mAP by more than 1.6%. In the future, we will focus on: 1) improving the scalability of the algorithm, to satisfy the demands of products categories dynamically changing with time in practice; and 2) extending the algorithm to other similar tasks by more explorations.

References

[1] H. Zhang, D. Li, Y. Ji, H. Zhou, W. Wu, and K. Liu, "Toward new retail: A benchmark dataset for smart unmanned vending machines," *IEEE Trans. Ind. Informat.*, vol. 16, no. 12, pp. 7722–7731, Dec. 2020.

[2] B. Hu, N. Zhou, Q. Zhou, X. Wang, and W. Liu, "DiffNet: A learning to compare deep network for product recognition," *IEEE Access*, vol. 8, pp. 19336–19344, 2020.

[3] Y. Zheng and Y. Li, "Unmanned retail's distribution strategy based on sales forecasting," in *Proc. 8th Int. Conf. Logistics, Informat. Service Sci. (LISS)*, Aug. 2018, pp. 1–5.

[4] H. Zhang, D. Li, Y. Ji, H. Zhou, and W. Wu, "Deep learning-based beverage recognition for unmanned vending machines: An empirical study," in *Proc. IEEE 17th Int. Conf. Ind. Informat. (INDIN)*, Jul. 2019, pp. 1464–1467.

[5] Y. Wei, S. Tran, S. Xu, B. Kang, and M. Springer, "Deep learning for retail product recognition: Challenges and techniques," *Comput. Intell. Neurosci.*, vol. 2020, pp. 1–23, Nov. 2020.

[6] L. Liu, B. Zhou, Z. Zou, S.-C. Yeh, and L. Zheng, "A smart unstaffed retail shop based on artificial intelligence and IoT," in *Proc. IEEE 23rd Int. Workshop Comput. Aided Modeling Design Commun. Links Netw. (CAMAD)*, Sep. 2018, pp. 1–4.

[7] C. Li *et al.*, "Data priming network for automatic check-out," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 2152–2160.

[8] L. Zhang, D. Du, C. Li, Y. Wu, and T. Luo, "Iterative knowledge distillation for automatic check-out," *IEEE Trans. Multimedia*, vol. 23, pp. 4158–4170, 2021.

[9] D. H. Kim, S. Lee, J. Jeon, and B. C. Song, "Real-time purchase behavior recognition system based on deep learning-based object detection and tracking for an unmanned product cabinet," *Expert Syst. Appl.*, vol. 143, Apr. 2020, Art. no. 113063.

[10] E. Goldman, R. Herzig, A. Eisenschtat, J. Goldberger, and T. Hassner, "Precise detection in densely packed scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5227–5236.

[11] X.-S. Wei, Q. Cui, L. Yang, P. Wang, and L. Liu, "RPC: A large-scale retail product checkout dataset," 2019, *arXiv:1901.07249*.

[12] Y. Bai, Y. Chen, W. Yu, L. Wang, and W. Zhang, "Products-10K: A large-scale product recognition dataset," 2020, *arXiv:2008.10545*.

[13] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[14] D. Li, H. Zhou, G. Li, B. Yang, F. Gao, and H. Zhang, "DrtNet: An improved RetinaNet for detecting beverages in unmanned vending machines," in *Proc. IEEE Int. Symp. Product Compliance Eng.-Asia (ISPCE-CN)*, Nov. 2020, pp. 1–6.

[15] L. Liu, J. Cui, Y. Huan, Z. Zou, X. Hu, and L. Zheng, "A design of smart unmanned vending machine for new retail based on binocular camera and machine vision," *IEEE Consum. Electron. Mag.*, vol. 11, no. 4, pp. 21–31, Jul. 2022.

[16] W. Liu, D. Anguelov, and, "SSD: Single shot multibox detector," in *Proc. ECCV*. Cham, Switzerland: Springer, 2016, pp. 21–37.

[17] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "DSSD: Deconvolutional single shot detector," 2017, *arXiv:1701.06659*.

[18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[19] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7263–7271.

[20] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[21] T.-Y. Lin, P. Goyal, and, "Focal loss for dense object detection," in *Proc. ICCV*, Oct. 2017, pp. 2980–2988.

[22] S. Ren, K. He, and, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[23] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.

[24] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 734–750.

[25] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6569–6578.

[26] X. Zhou, J. Zhuo, and P. Krahenbuhl, "Bottom-up object detection by grouping extreme and center points," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 850–859.

[27] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9627–9636.

[28] Z. Hu, J. Tang, and, "Deep learning for image-based cancer detection and diagnosis-a survey," *Pattern Recognit.*, vol. 83, pp. 134–149, 2018.

[29] Y. F. Said and M. Barr, "Pedestrian detection for advanced driver assistance systems using deep learning algorithms," *IJCSNS*, vol. 19, no. 9, p. 10, 2019.

[30] A. B. Mabrouk and E. Zagrouba, "Abnormal behavior recognition for intelligent video surveillance systems: A review," *Expert Syst. Appl.*, vol. 91, pp. 480–491, Jan. 2018.

[31] Y. Yang, F. Nie, D. Xu, J. Luo, Y. Zhuang, and Y. Pan, "A multimedia retrieval framework based on semi-supervised ranking and relevance feedback," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 723–742, Apr. 2011.

[32] P. Hou, X. Geng, and M.-L. Zhang, "Multi-label manifold learning," in *Proc. AAAI*. Princeton, NJ, USA: Citeseer, 2016, pp. 1680–1686.

[33] P. Zhao, G. Wu, S. Yao, and H. Liu, "A transductive transfer learning approach based on manifold learning," *Comput. Sci. Eng.*, vol. 22, no. 1, pp. 77–87, Jan. 2020.

[34] C. He, M. Tu, D. Xiong, and M. Liao, "Nonlinear manifold learning integrated with fully convolutional networks for PolSAR image classification," *Remote Sens.*, vol. 12, no. 4, p. 655, Feb. 2020.

[35] N. Pezzotti *et al.*, "GPGPU linear complexity t-SNE optimization," 2018, *arXiv:1805.10817*.

[36] R. Priam, "Symmetric generative methods and tSNE: A short survey," in *Proc. VISIGRAPP (IVAPP)*, 2018, pp. 356–363.

[37] D. Li and J. Yan, "3d shapes isometric deformation using in-tSNE," in *Proc. 8th Int. Conf. (ICIG)* (Lecture Notes in Computer Science), vol. 9217, Y. Zhang, Ed. Tianjin, China: Springer, Aug. 2015, pp. 1–9, doi: 10.1007/978-3-319-21978-3_1.

[38] J. Wehrmann, R. Cerri, and R. Barros, "Hierarchical multi-label classification networks," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 5075–5084.

[39] J. Yu, C. Zhu, J. Zhang, Q. Huang, and D. Tao, "Spatial pyramid-enhanced NetVLAD with weighted triplet loss for place recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 2, pp. 661–674, Feb. 2020.

[40] G. Wang, Y. Yuan, X. Chen, J. Li, and X. Zhou, "Learning discriminative features with multiple granularities for person re-identification," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 274–282.

[41] D.-X. Li, G.-Y. Fei, and S.-W. Teng, "Learning large margin multiple granularity features with an improved Siamese network for person re-identification," *Symmetry*, vol. 12, no. 1, p. 92, Jan. 2020.

[42] R. Yang, B. Ni, C. Ma, Y. Xu, and X. Yang, "Video segmentation via multiple granularity analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3010–3019.

[43] Z. Luo, J. Yuan, and, "Spatial constraint multiple granularity attention network for clothesretrieval," in *Proc. ICIP*, Sep. 2019, pp. 859–863.

[44] D. Wang, Z. Shen, J. Shao, W. Zhang, X. Xue, and Z. Zhang, "Multiple granularity descriptors for fine-grained categorization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2399–2406.

[45] J. Yu, M. Tan, H. Zhang, Y. Rui, and D. Tao, "Hierarchical deep click feature prediction for fine-grained image recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 2, pp. 563–578, Feb. 2022.

[46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[47] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *Proc. ICCV*, Oct. 2017, pp. 2961–2969.

[48] P. Bourke, "Capturing omni-directional stereoscopic spherical projections with a single camera," in *Proc. 16th Int. Conf. Virtual Syst. Multimedia*, Oct. 2010, pp. 179–183.

[49] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2921–2929.

[50] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 565–571.

[51] S. S. M. Salehi, D. Erdogmus, and, "Tversky loss function for image segmentation using 3D fully convolutional deep networks," in *Proc. Int. Workshop Mach. Learn. Med. Imag.*, Cham, Switzerland: Springer, 2017, pp. 379–387.

[52] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *IEEE CVPR*, Jul. 2017, pp. 4700–4708.

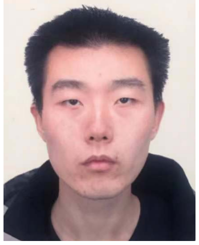
[53] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1492–1500.

Chengxu Liu received the B.E. degree from Xi'an Jiaotong University, Xi'an, China, in 2019, where he is currently pursuing the Ph.D. degree with the SMILES Laboratory.

His current research interests include fine-grained image classification, object detection, video super-resolution, and video frame interpolation.

Zongyang Da received the B.E. degree from Xi'an Jiaotong University, Xi'an, China, in 2020, where he is currently pursuing the M.E. degree with the SMILES Laboratory.

His current research interests include single-image super-resolution, blind super-resolution, and object detection.



Yuanzhi Liang received the B.E. degree from Lanzhou University, Lanzhou, China, in 2017. He is currently pursuing the M.E. degree with the SMILES Laboratory, Xi'an Jiaotong University, Xi'an, China.

His current research interests include visual relationships, fine-grained image classification, and object detection.

Yao Xue received the B.S. degree from the Xi'an University of Posts and Telecommunications, Xi'an, China, in 2010, the M.S. degree from Xi'an Jiaotong University, Xi'an, in 2013, and the Ph.D. degree from the University of Alberta, Edmonton, AB, Canada, in 2018.

He is currently a Lecturer with Xi'an Jiaotong University. His research interests include computer vision, medical image analysis, machine learning, and artificial intelligence.

Guoshuai Zhao (Member, IEEE) received the B.E. degree from Heilongjiang University, Harbin, China, in 2012, and the M.S. and Ph.D. degrees from Xi'an Jiaotong University, Xi'an, China, in 2015 and 2019, respectively.

He was an Intern with the Social Computing Group, Microsoft Research Asia, Beijing, China, from January 2017 to July 2017. He was a Visiting Scholar with Northeastern University, Boston, MA, USA, from October 2017 to October 2018, and MIT, Cambridge, MA, USA, from June 2019 to December 2019. He is currently an Assistant Professor with Xi'an Jiaotong University. His research interests include social media big data analysis, recommender systems, and natural language generation.

Xueming Qian (Member, IEEE) received the B.S. and M.S. degrees from the Xi'an University of Technology, Xi'an, China, in 1999 and 2004, respectively, and the Ph.D. degree in electronics and information engineering from Xi'an Jiaotong University, Xi'an, in 2008.

From 2011 to 2014, he was an Associate Professor with Xi'an Jiaotong University, where he is currently a Full Professor and the Director of the SMILES Laboratory. He was a Visiting Scholar with Microsoft Research Asia, Beijing, China, from 2010 to 2011. His current research interests include social media big data mining and search.

Prof. Qian was a recipient of the Microsoft Fellowship in 2006 and the Outstanding Doctoral Dissertations of Xi'an Jiaotong University and Shaanxi Province in 2010 and 2011, respectively.