2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM)

Fashion Product Search Based on Multi-model Fusion and Re-rank

Wenxiang Shang
Hangzhou Zhiyi Technology Co., Ltd.
Hangzhou, China
wenxiangshang@zhiyitech.cn

Feng Hu
Hangzhou Zhiyi Technology Co., Ltd.
Hangzhou, China
zhiyi_hufeng@163.com

***Abstract*—Fashion products search is an important pillar of the online store business. By analyzing the photos of fashion products taken by users, the high-precision and high-efficiency retrieval of the corresponding products in the huge fashion products database is a technology with broad application scenarios, which can help photo shopping, personalized recommendation, A number of artificial intelligence products such as advertising click prediction. At the same time, the angle, illumination, occlusion, size, quality, etc. of the products in the query image pose great challenges to the problem. In this paper, we use yolo-v3 to pre-crop the image of the product. We use a deep learning model to extract image features and calculate the distance between images. The results of the multiple trainings are then fused to obtain the final result.**

***Index Terms*—Fashion products search, Fusion, Re-rank**

I. Introduction

Fashion products search is a task of establishing correspondences between product images taken by users and images of a product provided by the merchant.

This task is very similar to the pedestrian re-identification task. Person re-identification is the task of establishing correspondences between observations of the same person in different videos and it has been widely applied in video surveillance. Person re-identification aims to re-identify a query person across multiple non-overlapping cameras. The task is challenging since pedestrian images from different camera views suffer from large variations in poses, lightings and backgrounds.

These challenges also exist in fashion products search. Many earlier works solve those problem by dividing it into two separated parts: feature extraction and metric learning A large number of handcrafted features are designed to enhance the robustness of pedestrian images to pose, viewpoint and illumination changes. After the feature is extracted, metric learning is applied to learn a metric for the features so that the images of the same person are close while the ones of different pedestrians are far away from each other in the metric space.

In recent years, convolutional neural networks(CNNs) have achieved promising results on person re-identification due to their advantages on feature learning. Different from previous works, CNNs learn features and metrics jointly from data in an end-to-end manner. Then an embedding is learned to measure the similarities between images using Euclidean distance.

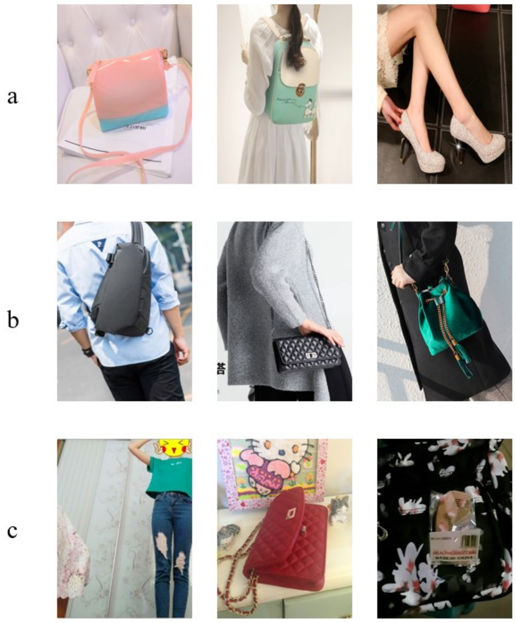

Figure.1 a is the original product image provided by the merchant, b is the cropped product image, and c is the user-uploaded captured image

In this paper, we use a similar process to end the fashion products search problem. But for the fashion products search task, there are some unique problems that need to be solved. In the fashion products search task, we focus on the goods in the image. As shown in Figure.1 (a), they generally only occupy a small part of the image, so the background has a great impact on the model. If we don't consider the background problem, it is difficult to solve this task well. So before we train the model, wo first pre-process the data. For each product image, we first use the target detection program to detect all the product boxes in the image, then use a box that covers all the boxes to represent the effective area of the image, and finally crop the target according to the effective area as the final training image.

As shown in Figure.1 (b) and (c), There is another problem that needs to be solved. (b) is an image of the product provided

978-1-5386-5321-0/18/$31.00 ©2018 IEEE

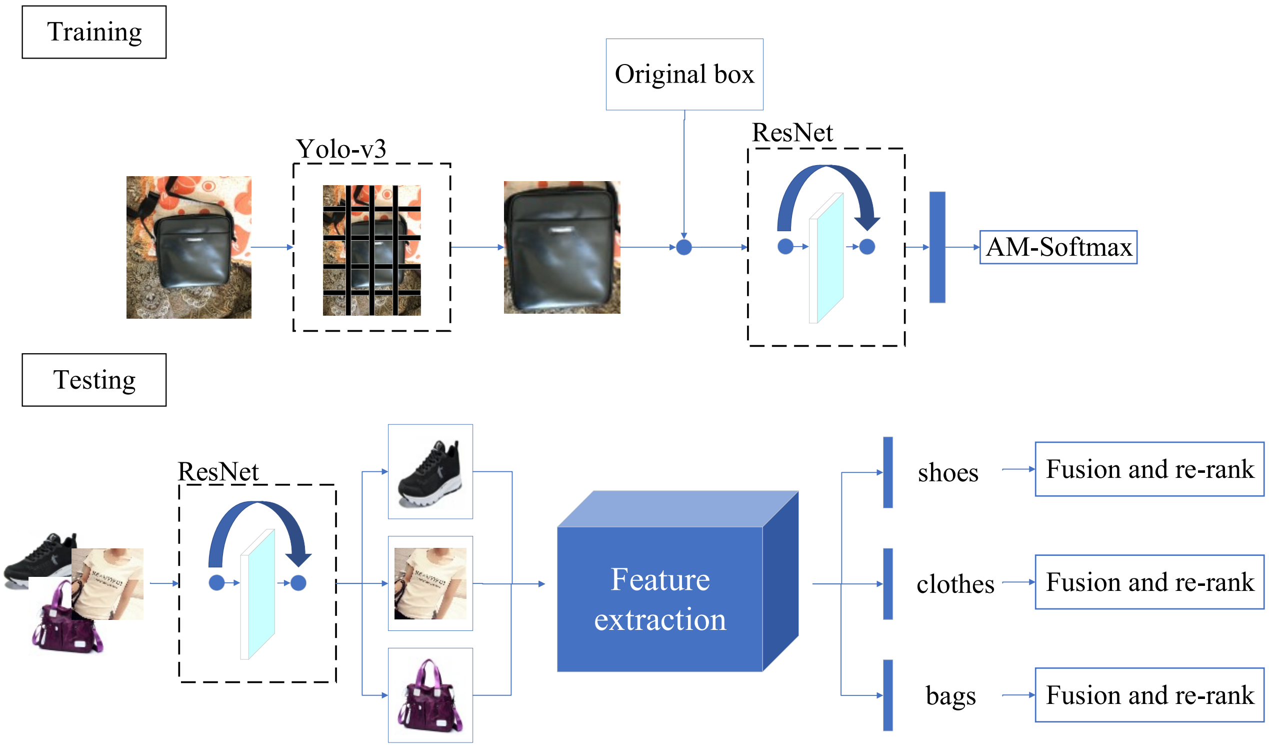

Figure.2 The overall structure of our model

by the merchant. Since the merchant has good shooting facilities and shooting conditions, the image clarity of these products is high, and the posture of the products is normal. The image in (c) is an image taken by the user himself and then uploaded. Users generally do not have good shooting equipment when shooting images. At the same time, users don't have good shooting skills when taking images, and the shooting conditions are poor. The image of the product photographed by the user usually has poor image quality, and the posture of the product is diverse. Therefore, combined with the above analysis, we can know that the image of the product photographed by the user and the image of the product photographed by the merchant have great differences in style, lighting conditions and posture. In response to these problems, after extracting the product image using the deep learning model, we specifically fuse and re-rank the distances between the image features. The specific fusion and re-ranking process will be described later in this article.

The remainder of this paper is organized as follows. Section 2 describes the data preprocessing approach and our training model framework. Section 3 details the distance fusion and re-rank methods. Section 4 summarizes the proposed algorithm.

II. Data Preprocessing and Training Model

A. Data Preprocessing

For each input image, we have the original crop box $[x_1, y_1, x_2, y_2]$. But as shown in figure.1(b), there are some wrong cropping boxes. Therefore, we use yolo-v3[1] to crop the original data to get a new crop box $[x_1', y_1', x_2', y_2']$. For the i-th input image, the final cropping box is represented as:

$$box_i = [\min(x_1, x_1'), \min(y_1, y_1'), \max(x_2, x_2'), \max(y_2, y_2')] \quad (1)$$

B. Training Model

After obtaining the final cropping boxes of the product images, we train a feature extraction model to extract the features of the products. We use resnet as the feature extraction model. For each product image, after the resnet model, we can get the feature F of the product. Then we use AM-Softmax[2] as a loss function to train the entire feature extraction model. For the three categories of bags, clothing, and shoes, we train the models separately to extract features. And for each category of model, we train four models of different final layer sizes (512,1024,2048,4096).

Since the images provided by the contestants are too small, we need to expand the training samples before training. For the input image I, height and width are w and h, respectively. Then we define the expansion of the image as:

$$x_{min}, y_{min} = Rand(0,0.15) * w, Rand(0,0.15) * h \quad (2)$$

$$x_{max}, y_{max} = Rand(0.85,1) * w, Rand(0.85,1) * h \quad (3)$$

$$I_{new} = I[x_{min}: x_{max}, y_{min}: y_{max}] \quad (4)$$

where $Rand(a, b)$ represents a random floatt number between a and b, I_{new} represents a new image with the same category. For each training image, we expand five new images in this way.

III. Feature Fusion and Re-rank

A. Classification

There are more than 150,000 images in the product search library. It is very difficult to search directly in the search library. Therefore, we classify the images in the search library before searching. We use Resnet50 as the classification model and cross entropy as the loss function.

B. Feature Extraction

For each query image, we first put the classification model into the category label. Then we use the previously trained feature extraction model to extract the features of the image. We define four different length features as: $f_{512}, f_{1024}, f_{2048}, f_{4096}$. We use the same method to extract the features of the images in the search library.

C. Fusion and Re-rank

After extracting the image features, we define the distance between the query image and the search library image as $\text{dist}(f^q, f^s)$. We fusion the distances of the four different length features extracted from each image:

$$dist(f^q, f^s) = \frac{1}{4}\left(dist\left(f^q_{512}, f^s_{512}\right) + dist\left(f^q_{1024}, f^s_{1024}\right) + dist\left(f^q_{2048}, f^s_{2048}\right) + dist\left(f^q_{4096}, f^s_{4096}\right)\right) \quad (4)$$

When sorting the final distance, we found that the image uploaded by the user is very different from the image of the product provided by the merchant. If we use the query image to search the library directly, we get a high top1 accuracy, but the top2-10 accuracy is very poor. Therefore, we first use the query image to find the top1 image in the search library. Then we use the top1 image to find the top2-10 image in the search library.

IV. Experimental Result

As shown in Table 1, when we did not use any optimization strategy, the map is 0.3544. It is very low. By observing, it can be found that the images in the top ten results returned by the search are very different. Then we fuse the feature distances of multiple models. The map has been raised to 0.4176. But it is still very low. After the final re-rank, the final map can reach 0.5375. The result styles of the top ten returned are similar. Therefore, fusion and re-rank are very important in the entire product search model.

TABLE I. Experimental Result of Different Strategies

Strategy	mAP
Original	0.3544
Fusion	0.4176
Fusion+Re-rank	0.5375

References

[1] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” arXiv preprint arXiv:1804.02767, 2018.

[2] F. Wang, W. Liu, H. Liu, and J. Cheng. Additive margin softmax for face verification. In arXiv:1801.05599, 2018.