IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Multi-Level Feature Network with Multi-Loss for Person Re-identification

HUIYAN WU, MING XIN, WEN FANG, HAI-MIAO HU, AND ZIHAO HU
Beijing Key Laboratory of Digital Media, School of Computer Science and Engineering, Beihang University, Beijing 100191, China

Corresponding Author: Hai-Miao Hu (frank0139@163.com)

This work was partially supported by the National Key Research and Development Program (Grant No.2016YFC0801003), the National Natural Science Foundation of China (No.61772058, 61421003), and the Fundamental Research Funds for the Central Universities.

ABSTRACT Person re-identification has become a challenging task due to various factors. One key to effective person re-identification is the extraction of the discriminative features of a person's appearance. Most previous works based on deep learning extract pedestrian characteristics from neural networks, but only from the top feature layer. However, the low-layer feature could be more discriminative in certain circumstances. Hence, we propose a method, named the Multi-Level Feature Network with Multiple Losses (MFML), which has a multi-branch network architecture that consists of multiple middle layers and one top layer for feature representations. To extract the discriminative middle-layer features and have a good effect on deeper layers, we utilize the triplet loss function to train the middle-layer features. For the top layer, we focus on learning more discriminative feature representations, so we utilize the Hybrid Loss (HL) function to train the top-layer feature. Instead of concatenating multilayer features directly, we concatenate the weighted middle-layer features and the weighted top-layer feature as the discriminative features in the testing phase. Extensive evaluations conducted on three datasets show that our method achieves a competitive accuracy level compared with the state-of-the-art methods.

INDEX TERMS Person re-identification, multi-level feature, multi-loss, deep learning

I. INTRODUCTION

Person re-identification (Re-ID) aims to establish correspondences among observations of the same person across non-overlapping cameras and short temporal periods. Person Re-ID has been drawing much attention and has played an important role in many practical applications [1-5, 10-12], for example, it can save a lot of human and material resources while in a large scene. However, it is still a challenge due to several uncontrolled complicated environment factors, such as time-varying light conditions, human pose changes and partial occlusions.

Discriminative feature representation is one of the important issues in person Re-ID. However, it is hard to compute discriminative biometric traits such as fingerprints, iris, face, gait, *etc*. due to some environmental constraints, such as the poor resolution of images [39]. Therefore, the person's visual appearance (e.g. clothing, skin color, *etc*.) may be the most discriminative features for the person Re-ID task [40], which means that an individual's visual appearance will not have noticeable changes in the person Re-ID task.

Recently, many studies concerning feature representation have been proposed, which can be roughly divided into two categories, namely, hand-crafted methods [1,2,3,17,18] and learning-based methods [4,5,10,11,12].

The hand-crafted methods are usually used to represent appearance features such as color, light, texture and combinations thereof [1,2,3]. The SDALF approach [1] focuses on three different visual characteristics of human appearance, including chromatic, structural and recurrent high-entropy textural characteristics. LOMO representation [2] uses local maximal occurrence features which implement the multiscale Retinex algorithm and apply the scale invariant local ternary pattern [37]. The HIPHOP method [3] extracts features from feature maps generated by a pre-trained network. It should be noted that hand-crafted features are deeply limited by practical constraints, although some improved hand-crafted features perform better than the original hand-crafted features.

Learning-based methods can outperform hand-crafted methods due to the powerful feature representation abilities offered by deep learning networks. Therefore, the current research focuses on extracting learned features from deep neural networks [4,5,10,11,12]. The GLAD method [4] first uses a convolutional neural network to divide a pedestrian image into three parts: head, upper-body and lower-body. Then, the global image and local images are used as input to the neural network which extracts the global and local features. McLaughlin et al. [5] utilize a recurrent neural network to extract additional useful information from consecutive video frames. Learned features often represent the high-level features of images, which achieves a higher accuracy in person re-identification [14, 41, 42].

Most existing person Re-ID methods based on deep learning extract features from the top level of the trained network because these features are strongly discriminative. However, a deep

neural network consists of multiple feature extraction layers, and the visual semantics of feature maps become more abstract when moving from the bottom to the top layers (see Fig. 1).

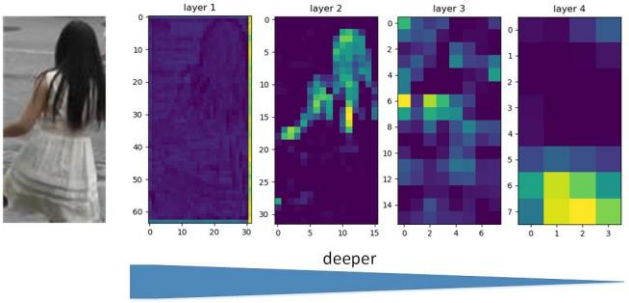


FIGURE 1. **Feature maps captured from multiple depth layers.**

Although the top-layer feature is more discriminative in many cases, the bottom-layer feature may perform better in certain cases. As illustrated in Fig. 2, we choose two pedestrian images (as shown in Fig. 2(a), Fig. 2(b)) and find the top 5 most similar images using the low-layer feature and top-layer feature. The top-layer feature performs better than the low-layer feature in Fig. 2(a), but the low-layer feature performs better than the top-layer feature in Fig. 2(b). We find empirically that different depth layer features represent different levels of semantics. Low-layer features represent low-level semantics, and top-layer features represent high-level semantics [16]. For the person Re-ID task, multilevel semantic features are part of the pedestrian feature representation. Motivated by this, we extract multilevel features from multiple layers of one deep neural network.

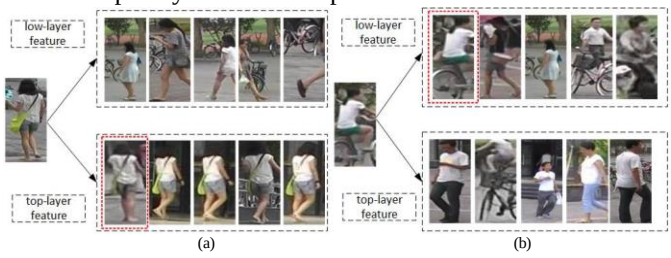


FIGURE 2. **In (a), the top 5 most similar images found through the top-layer feature are correct. However, the top 5 most similar images found through the low-layer feature are incorrect. In (b), the top 5 most similar images found through the low-layer feature are correct. However, the most similar top 5 images found through the top-layer feature are incorrect.**

In this paper, a neural network named the Multi-Level Feature Network with Multiple Losses (MFML) is proposed to enhance the feature representations from two aspects, the extraction of multi-level feature and the design of the loss function, which are shown in Fig. 4. To compensate for the insufficiency of the high-level semantic feature, this paper designs one multi-level network to generate multiple semantic features from multiple layers and combine them together. The feature extraction layers of this network are uniformly located at different depths of the backbone so that the features are uniformly distributed across the levels. In addition, to balance the contribution of each layer feature to the final feature, they are assigned the same average value through data transformation before combining them. For the loss function, we use the Hybrid Loss (HL) [8] at the end of the network to first ensure the representation ability of the top-layer feature. Moreover, we utilize a triplet loss function behind each middle feature extraction layer to improve the representation ability of the low-level features. The triplet loss function can enlarge the distance between interclass features and reduce the distance between intra-class features, leading to more discriminative features. Additionally, the triplet loss function will not change the projection direction of the features. Finally, all of the features are combined into the final feature. The experimental results based on three public datasets demonstrate that our MFML method has a competitive level of accuracy compared to that of the state-of-the-art methods. Furthermore, our method is suitable for different network depths and different types of basic networks.

In summary, the major contributions of this paper are the following. 1) Using the multilevel features of one network improves the robustness of the final feature. 2) Each selected layer is subject to one loss function, ensuring that the feature from each layer is strongly discriminative. 3) When we fuse all the layer features, the contribution of each layer feature to the final feature is balanced.

The rest of the paper is organized as follows. The second part reviews some of the related methods in the field of person re-identification. The third part introduces our proposed approach. The fourth part presents and analyzes the experimental results. Finally, we summarize the article.

II. RELATED WORK

Related convolutional networks: Some popular deep convolutional networks have been proposed in recent years, including AlexNet[9], ResNet[6], ResNeXt[36] and DenseNet [7]. Many works show that ResNet has better performance than other baseline networks on the person Re-ID task [19, 24]. Hence, ResNet is widely used. Most existing methods investigate the person Re-ID task, choosing a pretrained ResNet50 network as the baseline network. Consequently, this paper mainly employs the ResNet to introduce the proposed architecture. Moreover, this paper also chooses DenseNet as the baseline network in further analysis. ResNet is the original deep residual network, and ResNeXt and DenseNet are both improved versions of it. Different from traditional networks, ResNet first employs residual blocks which improve the accuracy of the network as the depth of the network increases. DenseNet strengthens the ResNet residual structure and deepens the common network. Since the network is deeper, it requires more accelerated graphics memory in the training phrase. However, its accuracy is better and the number of parameters is greatly reduced, which is an important indication for the person Re-ID task. ResNeXt utilizes group convolutions to divide the feature maps into groups which are concatenated together after the respective convolutions. Compared with ResNet, ResNeXt has higher accuracy and fewer parameters. These networks all have an obvious hierarchical structure, hence, it is convenient to divide them into several typical stages.

Feature fusion: Person re-identification algorithms have been studied for decades, and the characteristics used to describe individuals are always changing. The original methods use simple descriptors to describe the images, such as color histograms [18], texture histograms, and local binary patterns [17]. With the development of deep learning, the learned features and the attribute features are also utilized to describe the images. Then, some methods that fuse different features are proposed. Lejbølle et al. [14] first obtain low-level features, mid-level features and high-level features by using various existing

methods. They utilize the proposed late fusing scheme to fuse these features. The experimental results show that the accuracy of the fused features is improved. In addition to these methods that directly utilize other methods to obtain different features, there are also some methods that obtain different features in other ways. Spindle Net [15] combines the characteristics of each body part of the person in the image. It first obtains the head, upper body, lower body, left arm, right arm, left leg and right leg by segmenting the image. It finally obtains eight features plus the total body and fuses all of them together. It uses only one loss measure when training the network. In addition, there are some methods that directly fuse the outputs of different depth layers in the same neural network. The MLFN [16] algorithm extracts the features of different layers from one ResNeXt network. However, it fuses all the features in the training phase and finally trains the network with only one loss. This method does not connect the loss function after each selected layer and thus cannot guarantee that the features from different layers are distinguishable.

Deep metric learning: Traditional metric learning methods, such as XQDA[2] and KISSME[17], mainly focus on learning a Mahalanobis distance through machine learning methods. As deep learning is gradually applied to person Re-ID, the design of metric learning method is gradually transformed into the design of loss function. Cross-entropy loss and triplet loss are two commonly used loss functions. The cross-entropy loss function recognizes the task of person re-identification as a task of multi-label classification. Each pedestrian image has an independent label, and it does not consider the similarity relationships between different image samples. One pedestrian image is enough as the input of the network. However, the triplet loss function requires at least three pedestrian images in one batch. The purpose of the triplet loss function is to reduce the distance between two samples belonging to the same class and increase the distance between two samples belonging to different classes. In addition, several loss functions have been proposed in recent years [8, 13]. Chen et al. [13] propose a quadruplet loss function, which is designed based on the triplet loss function. Compared to triplet loss, quadruplet loss leads to the model output with a larger inter-class variation and a smaller intra-class variation. Hence, it has a better generalization ability and can achieve a higher performance level on a test dataset. The Hybrid Loss [8] is a new hybrid loss function that utilizes cross-entropy loss and triplet loss to learn the spatial distribution of features and distance between features more effectively.

III. PROPOSED APPROACH

A. THE MFML ARCHITECTURE

The overall model framework based on ResNet50 is illustrated in Fig. 4. Frameworks based on other networks are not significantly different from this one, and the experiments prove that the algorithm is effective for various basic networks.

ResNet[6] is a commonly used network in person re-identification based on deep learning, and it has an obvious hierarchical structure. Table 1 shows the architectures for ResNet34, ResNet50 and ResNet101 [6], in which each brackets represents a building block.

We use ResNet50 to illustrate the detailed architectures shown in Table 1. "conv1" contains only one convolution layer: 7×7, 64, stride 2. "conv2_x" first includes one 3×3 pooling layer and then contains 3 building blocks, each building block is denoted as:

$$\begin{bmatrix} 1\times 1,64 \\ 3\times 3,64 \\ 1\times 1,256 \end{bmatrix} \quad (1)$$

And each such building block contains three convolution layers, which is shown in Fig. 3. Therefore, "conv2_x" contains 3*3=9 convolutional layers. Similarly, "conv3_x" contains 4 residual modules, which means that it contains 3*4=12 convolutional layers. "conv4_x" contains 3*6=18 convolutional layers and "conv5_x" contains 3*3=9 convolutional layers. The last line of Table 1 contains a fully connected layer.

Therefore, ResNet is clearly divided into five stages: conv1, conv2_x, conv3_x, conv4_x, and conv5_x, which are represented by stage_0~stage_4, as shown in Fig. 4. Because there is only one convolution layer in stage_0, we combine it with stage_1 as the first stage. Therefore, stage_1~stage_4 are regarded as the selected stages.

TABLE 1. Architectures for ResNet34, ResNet50 and ResNet101 in this paper. Building blocks are shown in brackets, with the numbers of blocks stacked [6].

layer name	34-layer	50-layer	101-layer
conv1	7×7,64, stride 2		
conv2_x	3×3 max pool, stride 2		
	$\begin{bmatrix} 3\times 3,64 \\ 3\times 3,64 \end{bmatrix}\times 3$	$\begin{bmatrix} 1\times 1,64 \\ 3\times 3,64 \\ 1\times 1,256 \end{bmatrix}\times 3$	$\begin{bmatrix} 1\times 1,64 \\ 3\times 3,64 \\ 1\times 1,256 \end{bmatrix}\times 3$
conv3_x	$\begin{bmatrix} 3\times 3,128 \\ 3\times 3,128 \end{bmatrix}\times 4$	$\begin{bmatrix} 1\times 1,128 \\ 3\times 3,128 \\ 1\times 1,512 \end{bmatrix}\times 4$	$\begin{bmatrix} 1\times 1,128 \\ 3\times 3,128 \\ 1\times 1,512 \end{bmatrix}\times 4$
conv4_x	$\begin{bmatrix} 3\times 3,256 \\ 3\times 3,256 \end{bmatrix}\times 6$	$\begin{bmatrix} 1\times 1,256 \\ 3\times 3,256 \\ 1\times 1,1024 \end{bmatrix}\times 6$	$\begin{bmatrix} 1\times 1,256 \\ 3\times 3,256 \\ 1\times 1,1024 \end{bmatrix}\times 23$
conv5_x	$\begin{bmatrix} 3\times 3,512 \\ 3\times 3,512 \end{bmatrix}\times 3$	$\begin{bmatrix} 1\times 1,512 \\ 3\times 3,512 \\ 1\times 1,2048 \end{bmatrix}\times 3$	$\begin{bmatrix} 1\times 1,512 \\ 3\times 3,512 \\ 1\times 1,2048 \end{bmatrix}\times 3$
	average pool,2048-d fc		

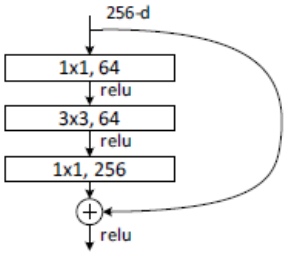


FIGURE 3. A BUILDING BLOCK IN "conv2_x" OF ResNet50

From the outputs of these selected stages, some middle-layer feature maps with a shape of Channel*Height*Width could be obtained. If these feature maps are directly used as the final feature of a pedestrian image, the calculation will be complicated. To address this problem, some branch structures with data transformations functions are constructed.

In one branch structure, a BatchNorm2d layer is firstly used to alleviate problems such as gradient disappearance and gradient explosion during the back propagation. And then a Relu layer is used to increase the nonlinear relationships among the layers of the network. An average pooling layer is utilized to change the shape of the feature map, which can reduce the computational load. The size of the pooling window is (Height,

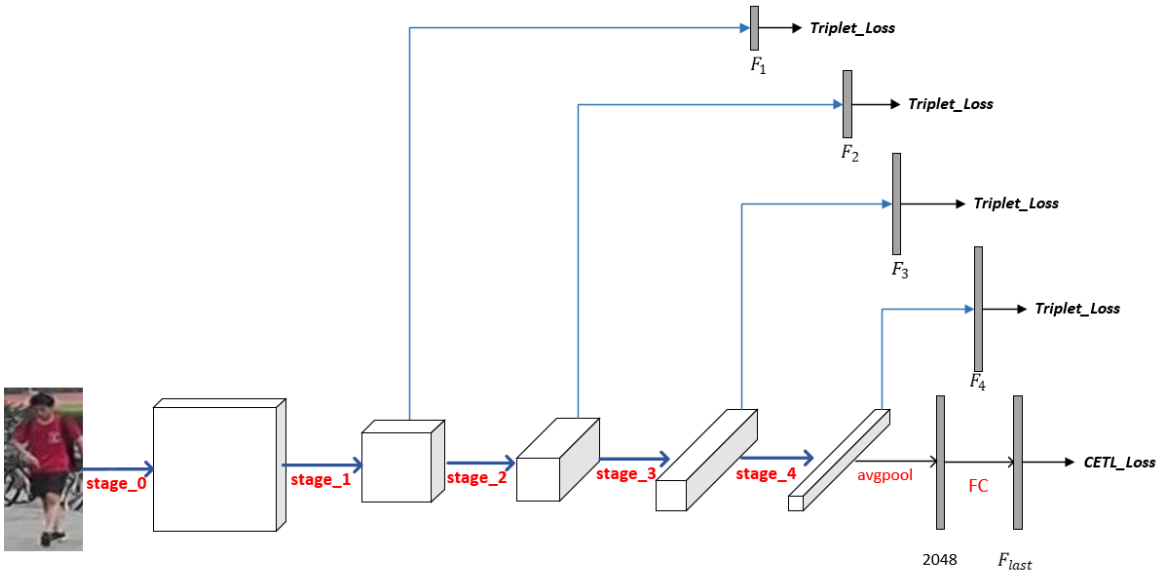


FIGURE 4. An overview of our proposed framework based on ResNet50. It consists of four branch blocks behind four stages, and each branch block is subject to one triplet loss function. One HL function is used at the end of the network.

Width), so the shape of a feature map becomes Channel*1. Finally, each middle-layer feature of each pedestrian image is transformed to:

$$F_x \in R^{C_x}, \tag{2}$$

where F_x is a middle-layer feature, C_x is the length of F_x. F_x can be F_1, F_2, F_3 or F_4 herein.

ResNet50 can output one 2048-dimensional feature for pedestrian images. Despite the fact that the feature length is different in different basic networks, the method is the same. And the feature does not need to be processed any more, which is denoted as F_{last} shown in Fig. 4.

B. FEATURE CONSTRUCTION

When testing the network, the final pedestrian characteristics are calculated using the five features, F_1, F_2, F_3, F_4 and F_{last}. The five features represent the characteristics of the pedestrians at different semantic levels. To make full use of them, this paper will cascade them together as the final pedestrian feature. However, if the data of one layer feature is too large or too small, its effect on the final feature will also be too large or too small. Therefore, to balance the contribution of each layer feature to the final feature, we will adopt an appropriate transformation to make the average of each layer feature vector almost equal.

We first calculate the average, w_x, of each layer feature in the test dataset:

$$w_x = \frac{1}{N_1}\sum_{i}^{N_1}\frac{1}{N_2}\sum_{j}^{N_2}F_x^{i\_j} \tag{3}$$

where N_1 is the number of samples in the test dataset, N_2 is the length of the vector F_x, and $F_x^{i\_j}$ is the j-th value in the feature vector F_x of the i-th sample. F_x can be $F_1, F_2, F_3, F_4, F_{last}$.

For each sample, its each layer feature is divided by the corresponding average value so that the average value of each layer feature is approximately 1, which brings a good balance to the contribution of each layer feature to the final result. Finally, we obtain the final feature F_{final} by connecting the layer features in series.

$$F_{final} = \text{concat}(\frac{F_1}{w_1}, \frac{F_2}{w_2}, \frac{F_3}{w_3}, \frac{F_4}{w_4}, \frac{F_{last}}{w_{last}}) \tag{4}$$

When the final feature of each sample is obtained, the Euclidean distance matrix between the query set and the gallery set can be calculated. Furthermore, the k-reciprocal coding [19] re-ranking algorithm is used to improve the re-identification accuracy. In addition to the accuracy advantage of this algorithm, it does not need additional information such as labels. It only relies on the original distances among the pedestrian images.

C. LOSS FUNCTION

1) TRIPLET LOSS

A triplet loss function is used after each middle stage mainly because of two valuable characteristics. First, it can enlarge the distance between the inter-class features and reduce the distance between intra-class features. Second, it will not change the projection direction of the middle-layer features so it will not have a bad influence on deeper layers.

F_1 is the first stage feature, and L_1 is the first stage loss. The similarity between two features is defined by their Euclidean distance. The calculation for the other stages is the same.

$$dist(F_1^i, F_1^j) = ||\ F_1^i, F_1^j\ ||_2 \tag{5}$$

where F_1^i is the stage_1 feature of the i-th pedestrian image.

Instead of using the traditional triplet loss function directly, we use hard sample mining as in [20]. The experimental results show that this improved triplet loss function performs better than the traditional triplet loss function. For each sample, the most similar sample with a different identity and the most dissimilar sample with the same identity are used to obtain the loss. The triplet loss of stage_1, L_1, is formulated as follows:

$$L_1 = \frac{1}{N}\sum_{i}^{N}[\max(\overbrace{\text{dist}\left(F_1^i, F_1^{i\_p}\right)}^{all\ postive\ pair}) - \min(\overbrace{\text{dist}\left(F_1^i, F_1^{i\_n}\right)}^{all\ negative\ pair}) + \alpha]_+ \quad (6)$$

where N is the batch size, α is the threefold threshold, $\text{dist}\left(\text{F}_1^{\text{i}}, \text{F}_1^{\text{i\_p}}\right)$ is the distance matrix of F_1 between the *i*-th pedestrian image and other images in the same category, and $\text{dist}\left(F_1^i, F_1^{i\_n}\right)$ is the distance matrix of F_1 between the *i*-th pedestrian image and the other images that do not belong to the same person. The improved triplet loss function not only improves the re-identification accuracy but also greatly reduces the computational load.

2) THE HYBRID LOSS

The Hybrid loss function is divided into two parts: cross-entropy loss and triplet loss. It actually combines the advantages of the two loss functions and makes the last layer feature, F_{last}, more discriminative. Its schematic is shown in Fig. 5.

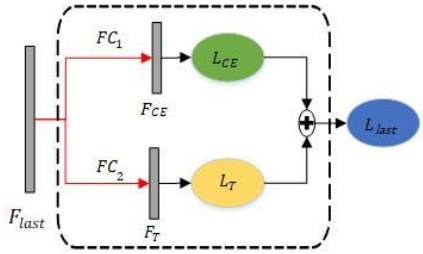


FIGURE 5. **An illustration of the Hybrid Loss.**

The F_{last} feature in Fig. 5 is the same as the F_{last} feature in Fig. 4, and L_{last} is the hybrid loss. The overall framework of the Hybrid Loss function is shown in the dotted box. It first obtains two feature vectors, F_{CE} and F_T, through two fully connected layers, FC_1 and FC_2. Then, the cross-entropy loss, L_{CE}, and the improved triplet loss, L_T, are calculated. Finally, the final hybrid loss, L_{last}, is obtained by L_{CE} plus L_T. The calculation of L_{last} is described in detail below.

The cross-entropy loss, L_{CE}, is calculated using the feature vector F_{CE}, and the formula is:

$$L_{CE} = -\frac{1}{N}\sum_{i}^{N} log\frac{e^{F_{CE}^i[label]}}{\sum_j^{N_2} e^{F_{CE}^i[j]}} \quad (7)$$

where the log in the formula is based on e and N is the batch size. N_2 is the length of the feature vector, F_{CE}, which is generally greater than or equal to the number of pedestrians in the training dataset. $F_{CE}^i[j]$ is the j-th value of the i-th sample in F_{CE}. The label indicates the category to which the pedestrian image belongs, that is, the id of the pedestrian.

The triplet loss, L_{T}, is calculated using the feature vector, F_T. Its calculation method is the same as that of the triplet loss for the middle layers, both of which use the improved triplet loss function. The formula is as follows:

$$L_{\text{T}} = \frac{1}{N}\sum_{i}^{N}[\max(\overbrace{\text{dist}\left(F_{\text{T}}^i, F_{\text{T}}^{i\_p}\right)}^{all\ postive\ pair}) - \min(\overbrace{\text{dist}\left(F_{\text{T}}^i, F_{\text{T}}^{i\_n}\right)}^{all\ negative\ pair}) + \alpha]_+ \quad (8)$$

where N is the batch size and α is the threefold threshold. The set of Euclidean distances between the F_T feature of the i-th pedestrian image and the other images of the same category is $\text{dist}\left(F_{\text{T}}^i, F_{\text{T}}^{i\_p}\right)$, and $\text{dist}\left(F_{\text{T}}^i, F_{\text{T}}^{i\_n}\right)$ is the set of Euclidean distances between the *i*-th pedestrian image and other images that do not belong to the same person.

The final hybrid loss is the weighted sum of the cross-entropy loss L_{CE} and the triplet loss L_T. Because we use the same weight for these two losses, the hybrid loss is the sum of the two losses:

$$L_{last} = L_{CE} + L_T \quad (9)$$

3) THE FINAL LOSS

When training the network, we first use the five features F_1, F_2, F_3, F_4 and F_{last} to obtain the five losses, as shown in Fig. 4. The middle-layer loss function used here is the improved triplet loss function. The main reason for using the triplet loss function is that it has good characteristics for the optimization of the middle layer. The triplet loss function can enlarge the distance between different classes and reduce the distance between the same class. Additionally, it does not change the original position of the feature in the feature space, so the optimization of this layer will never negatively affect the deeper network layers. Therefore, the triplet loss function is effective for the optimization of the middle layer. The four middle-layer losses L_1, L_2, L_3, L_4 are obtained and, finally, we use the Hybrid Loss function behind the last layer. The Hybrid Loss function is a new loss function that fuses the cross-entropy loss and triplet loss. It can improve the discrimination of the features and thus obtain L_{last}.

Finally, we obtain the final loss, L_{final}, by using the five loss values calculated before:

$$L_{final} = L_1 + L_2 + L_3 + L_4 + L_{last} \quad (10)$$

D. SUMMARY OF THE PROPOSED METHOD

To compensate for the insufficiency of the top layer feature, this paper designs one multi-branch network to generate multi-level feature. Its overview is shown in the Fig. 4. Generally, our method is built on one convolution neural network, such as ResNet and DenseNet. And we choose some feature extraction layers from the network according to the characteristics of the network.

When training the model, we use a triplet loss function behind each middle-layer to enhance the feature representation of each layer features. The triplet loss can enlarge the distance between interclass features and reduce the distance between intra-class features. In addition, the triplet loss function will not change the projection direction of features. At the same time, we use the Hybrid Loss function at the end of the model. The Hybrid Loss is composed of one cross-entropy loss and one triplet loss. And the Hybrid Loss performs better on the top layer feature. Finally, we add all of the losses to obtain the final loss.

When testing the network, we obtain each layer features through data transformation and combine all of them. To balance the contribution of each layer feature to the final feature, we make them have the same average value through the data processing before combining them in series. Finally, we use the re-ranking algorithm to improve the accuracy of person re-identification.

IV. EXPERIMENTS AND RESULTS

A. EXPERIMENTAL SETTINGS

1) DATASETS DESCRIPTION

In our experiments, three public benchmarks, Market1501[21] , CUHK03 [23] and DukeMTMC-reID [24], are used to train and validate the proposed framework. In each of the three datasets, the period of time between instances of the same person in multiple camera streams is short, usually no more than one day.

Market1501[21] is a commonly used large benchmark dataset. It was collected in front of a supermarket at Tsinghua University. It contains 12,936 training images and 19,732 gallery images (including 2,793 distracting images) of 1501 labeled persons from six cameras. It is divided into 751 identities for training and 750 identities for testing. There are an average of 17.2 images per identity, with different appearances. All the bounding boxes are produced using DPM [22].

CUHK03[23] was released by Li et al. in 2014. The images were taken in an area of the Chinese University of Hong Kong campus. It is the first dataset that could be used for person re-identification based on deep learning. CUHK03 consists of 13,164 images of 1,467 identities. These images were captured by 6 cameras. There are two settings: manually labeled person images and person images labeled by a DPM detector [22], both of which we will use. In this paper, we randomly select 1367 identities as the training dataset, 100 as the validation dataset and 100 as the testing dataset.

The DukeMTMC-reID dataset [24] is the person Re-ID subset of the Duke Dataset [25]. The DukeMTMC dataset is a relatively large database that contains 85 minutes of high-resolution video. It was collected from 8 high-resolution cameras, and can be used as an experimental dataset in the field of pedestrian tracking. There are 16,522 training images comprised of 702 identities, 2,228 query images and 17,661 gallery images of the other 702 identities in the DukeMTMC-reID dataset. This dataset was constructed by randomly selecting the manually labeled tracking bounding boxes.

2) EVALUATION METRICS

For performance evaluation, we employ Cumulative Matching Characteristic (CMC) cure and mean Average Precision (mAP) that are widely used in the person ReID literatures.

The Rank-k accuracy is used in the CMC. Rank-k accuracy stands for the accuracy that the matched image in the gallery is included in the top-k answers based on the similarity score. The Rank-k accuracy R(k) is calculated as follows:

$$R(k) = \frac{1}{M}\sum_{i}^{M} P_i^k \tag{11}$$

where M is the number of query images. If there is an image belonging to the same person as the *i*-th image in the returned top-*k* images, P_i^k=1, otherwise P_i^k=0.

MAP is proposed by L. Zheng [21]. For each query, the area under the Precision-Recall curve is firstly calculated, which is known as Average Precision (AP). Therefore, the mAP is calculated as follows:

$$\text{mAP} = \frac{1}{M}\sum_{i}^{M} AP_i \tag{12}$$

where M is the number of query images. AP_i is the AP of the *i*-th image.

We apply Rank-1 and mAP in Market1501 and DukeMTMC-reID and apply CMC in CUHK03.

3) IMPLEMENTATION DETAILS

The deep learning framework used in this paper is PyTorch. Its main programming language is python. It supports the dynamic construction of neural networks. The basic network models, ResNet and DenseNet, used in this paper are both pre-trained model architectures that are included in the TORCHVISION package.

The optimization function used in this paper is Adam[26], and the number of iterations is set as 300. The learning rate, ε, is initially set as 0.0002. In the first 200 generations, the value of the learning rate is unchanged, which is always 0.0002. When the generation exceeds 200, the learning rate will change according to the following formula:

$$\epsilon \leftarrow \epsilon * 0.001^{\frac{N-200}{100}} \tag{13}$$

where $\mathrm{N}(\mathrm{N} \geq 200)$ is the number of generations.

In this paper, all the input pedestrian images are first unified to a size of 256*128, and all the images are randomly flipped horizontally to expand the dataset. Finally, all the images are standardized. The average value of the three channels is set as 0.485, 0.456, and 0.406. The standard deviation is set as 0.229, 0.224, 0.225.

The batch size in this paper is set as 128. The drop value in the last dropout layer is 0.2, and the threshold, α, used in the triplet loss function is 0.5. The shape of the fully connected layer before F_{last} is 2048*2048. In the Hybrid Loss function, the shape of the fully connected layer, FC_1, is 2048*numclass (the number of pedestrians in the dataset), and the shape of the fully connected layer, $FC_{2,}$ is 2048* 1024.

After each iteration, the new model is automatically saved, and the model with the highest accuracy on the validation dataset is saved. In the final testing phase, the saved model with the highest accuracy in 300 generations will be used.

B. COMPARISON TO THE STATE-OF-THE-ART METHODS

In Table 2, Table 3 and Table 4, we compare our methods against the state-of-the-art methods on three person Re-ID benchmarks. There are 14 methods used for comparison, including methods based on feature: Spindle [15], MSCAN [27], JLML [28], DuATM [29], HA-CNN [31], MaskReID [32], DLPA [33], GLAD [4], MLFN [16], PCB [38] and some other methods: SSM [34], LFLMH [14], READ [35], SVDNet-ResNet50 [30]. In particular, we want to compare our method with MLFN. MLFN proposes a similar idea to our method, which is to employ one multi-level network to extract features, but it is different from our method in the construction of the loss function and the choices of the backbone network of architecture. Note that to compare with other methods fairly, this paper chose ResNet50 as the basic model in this part. The accuracy rate is obtained after re-ranking on each of the previously mentioned databases, and the best scores are shown in bold.

1) COMPARISON ON THE MARKET-1501 DATASET

We evaluated our methods against 12 existing methods on the Market-1501 dataset, as shown in Table 2. Single-Query and Multi-Query correspond to the single and multiple query setting

respectively [21]. The table clearly shows that our proposed method out performs most of the state-of-the-art methods. Although PCB (feature aligned method) outperforms our method by 1.3% (93.8 – 92.5) in terms of Single-Query Rank1, our method outperforms it by 7.7% (89.3-81.6) in terms of Single-Query mAP.

TABLE 2. MATCHING RATES (%) USING MARKET-1501 (BEST SCORES ARE SHOWN IN BOLDFACE)

Methods	Single-Query		Multi-Query	
	mAP	Rank1	mAP	Rank1
SVDNet-ResNet50	62.1	82.3	-	-
SSM	68.8	82.2	-	-
REDA	71.31	87.08	-	-
Spindle	-	76.9	-	-
MSCAN	57.5	80.3	66.7	86.8
DLPA	63.4	81.0	-	-
JLML	65.5	85.1	74.5	89.7
GLAD	73.9	89.9	-	-
HA-CNN	75.7	91.2	82.8	93.8
DuATM	76.62	91.42	-	-
PCB	81.6	**93.8**	-	-
MaskReID	88.03	92.04	91.94	94.18
MLFN	74.3	90.0	82.4	92.3
Ours-ResNet50	**89.3**	92.5	**92.4**	**94.2**

2) COMPARISON ON THE CUHK03 DATASET

We evaluate the MFML network on the CUHK03 dataset against 8 existing methods. Because the CUHK03 dataset is divided into manual detection (labeled) and algorithmic detection (detected) images, according to the pedestrian detection method, the experiments were carried out on these two datasets, and the comparison results are shown in Table 3. It can be seen from the data in the table that the accuracy of our method is higher than that of the other state-of-the-art methods. Especially compared to the similar method, MLFN, our method has obvious advantages, as is the case on the other two datasets. And our method outperforms the 2^{nd} best model, Spindle (fusing local and global features), by 5.9% (94.4-88.5) in terms of Rank1.

TABLE 3. MATCHING RATES (%) ON THE CUHK03 (BEST SCORES ARE SHOWN IN BOLDFACE)

Methods	Labeled			Detected		
	Rank1	Rank5	Rank10	Rank1	Rank5	Rank10
LFLMH	54.72	81.25	89.40	-	-	-
SSM	76.6	94.6	98.0	72.7	92.4	96.1
HA-CNN	44.4	-	-	41.7	-	-
PCB	-	-	-	63.7	80.6	86.9
MSCAN	74.2	94.3	97.5	68.0	91.0	95.4
GLAD	85.0	97.9	99.1	82.2	95.8	97.6
DLPA	85.4	97.6	99.4	81.6	97.3	98.4
Spindle	88.5	97.8	98.6	-	-	-
MLFN	54.7	-	-	52.8	-	-
Ours-ResNet50	**94.4**	**99.1**	**99.6**	**91.6**	**97.7**	**98.6**

3) COMPARISON ON THE DUKEMTMC-REID DATASET

We evaluate our method on the DukeMTMC-ReID dataset, and the comparison results are shown in Table 4. This dataset is more challenging than the other two datasets. As shown in the table, the accuracy rate is relatively low. Our method outperforms the 2^{nd} best model, MaskReID, by 0.27% (80.0-79.73) in terms of mAP. Compared to MaskReID, our method only has a slight performance advantage, as is the case on Market1501. However, its idea and approach are different from ours. The main contribution of MaskReID is to design a network that takes both the original and the mask images as inputs. And we speculate that if mask images are also used in our method, the performance will increase.

TABLE 4. MATCHING RATE (%) ON THE DukeMTMC-ReID (BEST SCORES ARE SHOWN IN BOLD)

Methods	mAP	Rank1
SVDNet-ResNet50	56.80	76.70
REDA	62.44	79.31
JLML	56.4	73.3
PCB	69.2	83.3
MaskReID	79.73	**84.07**
MLFN	62.8	81.0
Ours-ResNet50	**80.0**	84.0

C. FURTHER ANALYSIS AND DISCUSSION

1) EVALUTION ON DIFFERENT NETWORK DEPTHS

In fact, our method is applicable to ResNet with different depths, not only ResNet50. Therefore, we will explore the performance of MFML on ResNet with different depths through experiments.

In this part, ResNet34, ResNet50, and ResNet101 are selected as the backbone network, and the experiments before and after the improvements are performed separately. The baseline experiment uses one ResNet with a Hybrid Loss function. The four graphs in Fig. 6 show the relationship between the accuracy before re-ranking and network depth under four different evaluation criteria. The four evaluation criteria are the mAP and Rank1 evaluation criteria in single-query mode and multi-query mode.

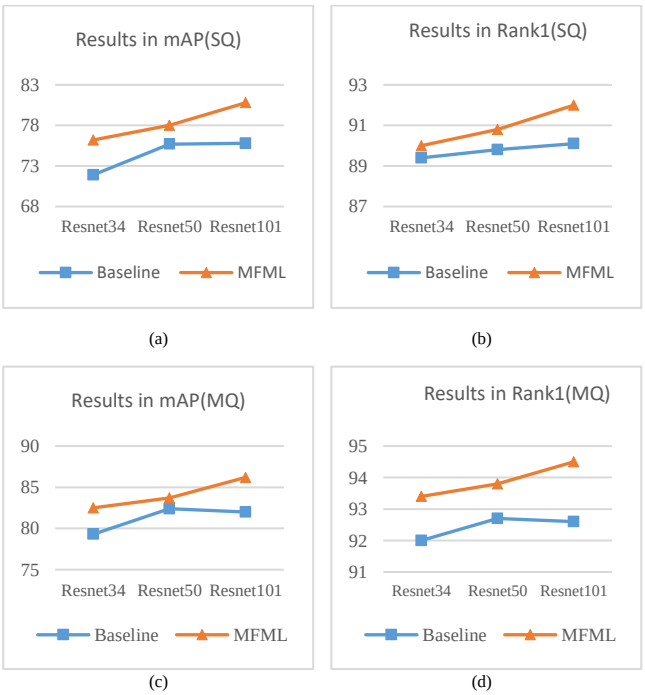


FIGURE 6. Comparison of accuracy for the baseline and MFML models. The four illustrations represent four evaluation criteria. In each illustration, the x-coordinate represents ResNet of different depths, and the y-coordinate represents the matching rate. The blue line represents the baseline, and the orange line represents our MFML model before re-ranking.

We can draw the following conclusions from Fig. 6:

(i) Regardless of the depth of the basic network, the accuracy of the MFML model is always higher than the accuracy of the baseline model under any evaluation criteria.

(ii) The original ResNet model has a good characteristic that its Re-id accuracy will increase as the depth of the network increases. As we can see from these four illustrations, the MFML model does not destroy the characteristics of the ResNet model.

(iii) The effect of our proposed method may be better than simply deepening the network. For example, in (b), the accuracy of the improved ResNet50 model is higher than that of the basic ResNet101 model.

2) DIFFERENT BASELINE MODEL

To prove that the effectiveness of our proposed method on the ResNet model is not accidental, we also perform experiments using DenseNet as the backbone network of our proposed method. Because of the hardware limitations, we only perform the experiment with a depth of 121. The baseline model is one DenseNet121 network with one Hybrid Loss function and the results before re-ranking are shown in Table 5. The upper line is the accuracy of the baseline model. The following line is the accuracy of the method proposed in this paper. It is obvious that our proposed method is effective regardless of which evaluation metric is used.

TABLE 5. MATCHING RATES (%) ON THE MARKET-1501 WHEN THE BACKBONE ARCHITECTURE IS DENSENET121

	Single-Query		Multi-Query	
	mAP	Rank1	mAP	Rank1
Densetnet121(Baseline)	75.0	89.9	81.6	92.7
Densetnet121(MFML)	78.5	91.0	84.2	93.6

Now the backbone network used in this paper include ResNet and DenseNet with different depths. All of these networks have a thing in common, that is, they all have obvious hierarchical structures. Each network is clearly divided into several stages and the number of stages is relatively small, usually 4~5satges. Therefore, we do not need to discuss the automatic feature selection for these networks in this paper.

However, if a network that does not have an obvious hierarchical structure is used as the backbone of the proposed MFML architecture, the lack of automatic feature selection will be a limitation of our method. In the future, we will refer to some methods [16, 43] which can select deep features to make our method applicable to more backbone networks.

3) EFFECT OF THE HYBRID LOSS

The Hybrid Loss (HL) is used to train the top-layer, which is introduced in detail in the Section III C 2).

In order to demonstrate the effect of HL on person re-identification, we design an experiment. We set the loss function followed by the top-layer to HL and triplet loss respectively, and then compare their performance. The backbone used in this experiment here is ResNet50 and the dataset is Market1501. The experimental results are shown in Table 6.

As can be seen from the Table 6, HL has obvious performance advantages regardless of the evaluation metric.

TABLE 6. EFFECT OF THE HL ON THE MARKET-1501. *BACKBONE:* ResNet50

	Single-Query		Multi-Query	
	mAP	Rank1	mAP	Rank1
MFML +triplet loss	85.0	89.3	89.2	92.0
MFML+HL	89.3	92.5	92.4	94.2

4) COMPUTATIONAL COMPLEXITY AND TIME COST

Complexity and time cost are also important evaluation criteria for model performance. In order to demonstrate the performance of our method, we compare the proposed MFML model (ResNet50-based) with its backbone ResNet50 and deeper network ResNet101 in complexity and time cost. In Table 7, we report the results of running the models with 256 * 128 * 3 images with a batch size of 64 under the environment of 4 NVIDIA Tesla M10 GPUs (8GB/GPU). "FLOPs" is the number of Floating-point Operations. "Training time" reflects the duration of a pair of forward and backward passes, averaged over 460 runs.

TABLE 7. COMPUTATIONAL COMPLEXITY AND TIME COST

	ResNet50	MFML(ResNet50-based)	ResNet101
FLOPs	2.7214×10^9	2.7254×10^9	5.1613×10^9
Parameter Number	3.4931×10^7	3.4939×10^7	5.3923×10^7
Training time(s)	0.721	1.007	1.092

Compared with the backbone network ResNet50, the FLOPs and Parameter Number of MFML almost have no increase, and the Training time only increases by 0.286s (1.007-0.721). Therefore, the complexity and time cost of MFML do not increase significantly, while the accuracy of MFML is significantly improved.

Compared with ResNet101, MFML has significantly fewer FLOPs and Parameter Number. And the Training time of MFML is 0.085s (1.092-1.007) less than that of ResNet101. However, the accuracy of MFML (ResNet50-based) is higher than that of ResNet101, which can be seen from Fig. 6.Therefore, this also proves the effectiveness of the proposed MFML in this paper.

V. Conclusion

In this paper, we propose a model named the MFML. In contrast to most existing deep learning-based re-identification models that only use the output of the top layer of the network as the pedestrian feature, the proposed model uses the features from multiple layers of one network. The features obtained from the different levels of the network correspond to the latent attributes of different semantic levels, which has been proven in other computer vision research. Furthermore, our proposed model connects a loss function behind each layer to enhance the discrimination of each layer feature. The experimental results prove the advantages of the proposed algorithm when compared with other state-of-the-art methods. Finally, we also discussed the impact of network depth and the use of different basic networks on the proposed model. In the future, we may further study the selection of the feature extraction layers.

REFERENCES

[1] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2010, pp.2360-2367.

[2] S. Liao, Y. Hu, X. Zhu, and S. Z. Li, "Person re-identification by local maximal occurrence representation and metric learning," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2015, pp. 2197–2206.

[3] Y. Chen, X. Zhu, W. Zheng and J. Lai, "Person re-identification by camera correlation aware feature augmentation," *IEEE Trans. Pattern Anal. Mach. Intell*., vol. 40, no. 2, pp. 392-408, Feb. 2018.

[4] L. Wei, S. Zhang, H. Yao, W. Gao, and Q. Tian, "GLAD: Global-localalignment descriptor for pedestrian retrieval," in *Proc. ACM Multimedia Conf.*, 2017, pp. 420–428.

[5] N. McLaughlin, J. M. d. Rincon and P. Miller, "Recurrent convolutional network for video-based person re-identification," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 1325-1334.

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

[7] G. Huang, Z. Liu, L. v. d. Maaten and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 2261-2269.

[8] Z. Hu, H. Wu, S. Liao, H. Hu, S. Liu and B. Li, "Person re-identification with hybrid loss and hard triplets mining," in *IEEE Int. Conf. Multi. Big. Data. (BigMM)*, 2018, pp. 1-5.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1106–1114.

[10] X. Zhang, H. Luo, X. Fan, W. Xiang, Y. Sun, Q. Xiao, W. Jiang, C. Zhang, and J. Sun.(2017). "AlignedReID: Surpassing human-level performance in person re-identification.'' [Online]. Available: https://arxiv.org/abs/1711.08184

[11] F. Tan, X. Zhao, K. Liu and Q. Liao, "Person Re-Identification across Non-Overlapping Cameras Based on Two-Stage Framework," in *Int. Conf. Mea. Tec. Mec. Auto. (ICMTMA)*, 2018, pp. 235-238.

[12] L. Lin, D. Liu, X. Li, F. Zhang and M. Ye, "Person re-identification based on viewpoint correspondence pattern," in *Int. Comput. Conf. Wav. Act. Media Tec. Inf. Proc. (ICCWAMTIP)*, 2017, pp. 116-119.

[13] W. Chen, X. Chen, J. Zhang and K. Huang, "Beyond triplet loss: A deep quadruplet network for person re-identification," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 1320-1329.

[14] A. R. Lejbølle, K. Nasrollahi and T. B. Moeslund, "Enhancing person re-identification by late fusion of low-, mid- and high-level features," in *IET Biometrics*, vol. 7, no. 2, pp. 125-135, 2018.

[15] H. Zhao, M. Tian, S. Sun, J. Shao, J. Yan, S. Yi, X. Wang and X, Tang, "Spindle net: Person re-identification with human body region guided feature decomposition and fusion," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 907-915.

[16] X. Chang, T. M. Hospedales, and T. Xiang. (2018). "Multi-level factorization net for person re-identification." [Online]. Available: https://arxiv.org/abs/1803.09132

[17] M. Koestinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2012, pp. 2288–2295.

[18] W. Li and X. Wang, "Locally aligned feature transforms across views," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2013, pp. 3594-3601.

[19] Z. Zhong, L. Zheng, D. Cao and S. Li, "Re-ranking person re-identification with k-reciprocal encoding," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 3652-3661.

[20] A. Hermans, L. Beyer, and B. Leibe. (2017). "In defense of the triplet loss for person re-identification." [Online]. Available: https://arxiv.org/abs/1703.07737

[21] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang and Q. Tian, "Scalable person re-identification: A benchmark," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 1116-1124.

[22] P. Felzenszwalb, D. McAllester and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2008, pp. 1-8.

[23] W. Li, R. Zhao, T. Xiao and X. Wang, "DeepReID: Deep filter pairing neural network for person re-identification," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2014, pp. 152-159.

[24] Z. Zheng, L. Zheng and Y. Yang, "Unlabeled samples generated by GAN improve the person re-identification baseline in vitro," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 3774-3782.

[25] E. Ristani, F. Solera, R. S. Zou, R. Cucchiara, C. Tomasi. (2016). "Performance measures and a data set for multi-target, multi-camera tracking." [Online]. Available: https://arxiv.org/abs/1609.01775

[26] D. Kingma, J. Ba.Adam. (2014) "A method for stochastic optimization." [Online]. Available: https://arxiv.org/abs/1412.6980

[27] D. Li, X. Chen, Z. Zhang, and K. Huang, "Learning deep context-aware features over body and latent parts for person re-identification," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 7398-7407.

[28] W. Li, X. Zhu, and S. Gong, ''Person re-identification by deep joint learning of multi-loss classification,'' in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2017, pp. 1–10.

[29] J. Si, H. Zhang, C. Li, J. Kuen, X. Kong, A. Kot, and G. Wang, "Dual Attention Matching Network for Context-Aware Feature Sequence based Person Re-Identification," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 5363-5372.

[30] Y. Sun, L. Zheng, W. Deng and S. Wang, "SVDNet for pedestrian retrieval," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 3820-3828.

[31] W. Li, X. Zhu, and S. Gong, "Harmonious Attention Network for Person Re-Identification," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 2285-2294.

[32] L. Qi, J. Huo, L. Wang, Y. Shi, and Y. Gao. (2018). "MaskReID: A Mask Based Deep Ranking Neural Network for Person Re-identication." [Online]. Available: https://arxiv.org/abs/1804.03864

[33] L. Zhao, X. Li, Y. Zhuang and J. Wang, "Deeply-learned part-aligned representations for person re-identification," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 3239-3248.

[34] S. Bai, X. Bai, and Q. Tian, "Scalable person re-identification on supervised smoothed manifold," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 3356-3365.

[35] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang. (2017). "Random erasing data augmentation." [Online]. Available: https://arxiv.org/abs/1708.04896

[36] S. Xie, R. Girshick, P. Doll´ar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp.5987-5995.

[37] S. Liao, G. Zhao, V. Kellokumpu, M. Pietikäinen, and S. Z. Li, "Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes," in *Proc. IEEE. Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 1301-1306.

[38] Y. Sun, L. Zheng, Y. Yang, Q. Tian and S. Wang, "Beyond Part Models: Person Retrieval with Refined Part Pooling (and A Strong Convolutional Baseline)," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 480-496.

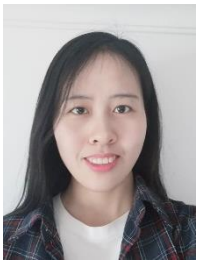
[39] A. Nanda, P. K. Sa, S. K. Choudhury, S. Bakshi and B. Majhi, "A Neuromorphic Person Re-Identification Framework for Video Surveillance," *IEEE Access*, vol. 5, pp. 6471-6482, 2017.

[40] N. Perwaiz, M. M. Fraz and M. Shahzad, "Person Re-Identification Using Hybrid Representation Reinforced by Metric Learning," *IEEE Access*, vol. 6, pp. 77334-77349, 2018.

[41] Q.Xiao, H.Luo and C.Zhang. (2017). "Margin Sample Mining Loss: A Deep Learning Based Method for Person Re-identification." [Online]. Available: https://arxiv.org/abs/1710.00478

[42] Y.Fu, Y.Wei, Yu.Zhou, H.Shi, G.Huang, X.Wang, Z.Yao and T.Huang. (2018). "Horizontal Pyramid Matching for Person Re-identification," [Online]. Available: https://arxiv.org/abs/1804.05275

[43] J.Nalepa, G.Mrukwa and M.Kawulok, "Evolvable Deep Feature," in *Applications of Evolutionary Computation*, vol.10784, pp. 497-505, 2018.



HUIYAN WU received the B.S. degree from Beihang University, Beijing, China, in 2018, and currently pursuing the M.S. degree from Beihang University, Beijing, China, in 2021, all in computer science. Her current research interests include person re-identification and object tracking.

MING XIN received her B.S. degree in information management and information system from Southwest University in 2002 and the M.S. in Applied Mathematics from Henan University, China, in 2008. She is currently a Ph.D. candidate at the School of Computer Science and Engineering, Beihang University, China. She joined the School of Computer and Information Engineering, Henan University in 2002, where she has been an Associate

Professor since 2013. Her current research interests focus on moving object detection and tracking, object recognition.

WEN FANG received the B.S. degree in computer science and technology from Wuhan University of Science and Technology, Hubei, China, in 2009, the M.S. Degree in mathematics and computer science from Fuzhou University, Fujian, China, in 2013, and she is currently pursuing the Ph.D. Degree in computer science and engineering from Beihang University, Beijing, China. Her current research interests include person re-identification, image enhancement, video analysis and understanding.

HAI-MIAO HU received the B.S. degree from Central South University, Changsha, China, in 2005, and the Ph.D. degree from Beihang University, Beijing, China, in 2012, all in computer science. He was a visiting student at University of Washington from 2008 to 2009.

Currently, he is an associate professor of Computer Science and Engineering at Beihang University. His research interests include video coding and networking, image/video processing, and video analysis and understanding.

ZIHAO HU received the B.S. degree from China University of Geosciences, Wuhan, China, in 2016, and the M.S. degree from Beihang University, Beijing, China, in 2019, all in computer science. His current research interests include video analysis and understanding.