Combining Pose Invariant and Discriminative Features for Vehicle Reidentification

Hao Sheng, *Member, IEEE*, Kai Lv, Yang Liu, Wei Ke, Weifeng Lyu, *Member, IEEE*, Zhang Xiong, and Wei Li, *Member, IEEE*

***Abstract*—Vehicle reidentification, aiming at identifying vehicles across images, has drawn a lot of attention and has made significant achievements in recent years. However, vehicle reidentification remains a challenging task caused by severe appearance changes due to different orientations. In practice, the result of reidentification is greatly influenced by the pose of vehicles, and we call this influence as a pose barrier problem. One way to address the pose barrier problem is to train a feature representation that is invariant for various vehicle poses. To this end, we present pose robust features (PRFs) that contains two components: 1) pose-invariant features (PIFs) and 2) pose discriminative features (PDFs). *On the one hand,* PIF is the expert in exploring the overall characteristic of vehicles. When training PIF, we adopt an identity classifier as well as an orientation classifier. In addition, an adversarial loss is deployed in the PIF network. *On the other hand,* we design a PDF network, which has a similar architecture to the PIF network but can distinguish the difference between local details. The difference between PDF and PIF is that the network of training PDF does not apply the adversarial loss. Finally, by combining PIF and PDF, PRF has the advantages of the two features and can alleviate the influence of the pose barrier problem. Experiments are conducted on the VeRi-776 and VehicleID data sets. We show that PIF and PDF are complementary and that PRF produces competitive performance compared with state-of-the-art approaches.**

***Index Terms*—Adversarial learning, image representation, pose discriminative, pose invariant, vehicle reidentification.**

Manuscript received May 9, 2020; revised June 24, 2020 and August 2, 2020; accepted August 4, 2020. Date of publication August 10, 2020; date of current version February 19, 2021. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB2100500; in part by the National Natural Science Foundation of China under Grant 61861166002, Grant 61872025, and Grant 61635002; in part by the Science and Technology Development Fund, Macau SAR, under Grant 0001/2018/AFJ; in part by the Fundamental Research Funds for the Central Universities; and in part by the Open Fund of the State Key Laboratory of Software Development Environment under Grant SKLSDE2019ZX-04. *(Corresponding author: Kai Lv.)*

Hao Sheng, Weifeng Lyu, Zhang Xiong, and Wei Li are with the State Key Laboratory of Software Development Environment, School of Computer Science and Engineering and the Beijing Advanced Innovation Center for Big Data and Brain Computing, Beihang University, Beijing 100191, China (e-mail: shenghao@buaa.edu.cn; lwf@buaa.edu.cn; xiongz@buaa.edu.cn; liwei@nlsde.buaa.edu.cn).

Kai Lv and Yang Liu are with the State Key Laboratory of Software Development Environment, School of Computer Science and Engineering, Beihang University, Beijing 100191, China, and also with the Beihang Hangzhou Institute for Innovation at Yuhang, Beihang University, Hangzhou 311121, China (e-mail: lvkai@buaa.edu.cn; liu.yang@buaa.edu.cn).

Wei Ke is with the School of Applied Sciences, Macao Polytechnic Institute, Macau, China (e-mail: wke@ipm.edu.mo).

Digital Object Identifier 10.1109/JIOT.2020.3015239

I. Introduction

In THIS work, we study the problem of vehicle reidentification between different cameras. Vehicle reidentification is a retrieval task that recognizes the same vehicle as they move through multiple cameras without overlapping views. Over the last decades, the Internet of Things (IoT) is receiving more and more attention from the research community. Connecting vehicle reidentification to IoT creates a powerful network capability. Being able to identify vehicles from cameras allows the local node to be more intelligent and have greater autonomy. In this way, vehicle reidentification can contribute to IoT by reducing the processing load on central servers and allowing a more distributed control architecture. Vehicle reidentification [1]–[3] serves wide applications in computer vision and multimedia fields, including surveillance and video retrieval. For example, surveillance systems are tasked with identifying a vehicle automatically instead of by human efforts. In these cases, the application of reidentification can help us find the objects in a faster and more accurate way.

In the process of identifying vehicles, vehicle pose or orientation plays a critical role that affects the identification accuracy. The influence of pose variations mainly exists in two aspects: 1) even with different identities, vehicles of the same pose are easily considered to be the same identity and 2) vehicles with the same identity may also be considered as different identities due to different poses. For example, as shown in the first row of Fig. 1(a), the probe vehicle and the first gallery vehicle have a similar silver appearance and the same pose. In this case, it is difficult to distinguish the identities of the vehicles. Fig. 1(b) shows the feature distance between the positive pair and the negative pair. As the vehicles of the positive pair have different poses, the distance of the positive pair is larger than the distance of the negative pair. In this article, we define the obstacle derived from different poses as the pose barrier problem. As shown in Fig. 1(c), in this work, we intend to learn a feature representation, where vehicles of the same identity are close to each other and the vehicles of different identities are far away. In this way, the proposed method can reduce the impact of vehicle poses on identification accuracy.

Some works have tried to solve the pose barrier problem for person reidentification [4]–[7] and vehicle reidentification [8]–[11]. For example, in tackling person reidentification, Cho and Yoon [4] got the pedestrian's several pictures of different poses through a picture sequence and calculated matching scores between poses in a weighted

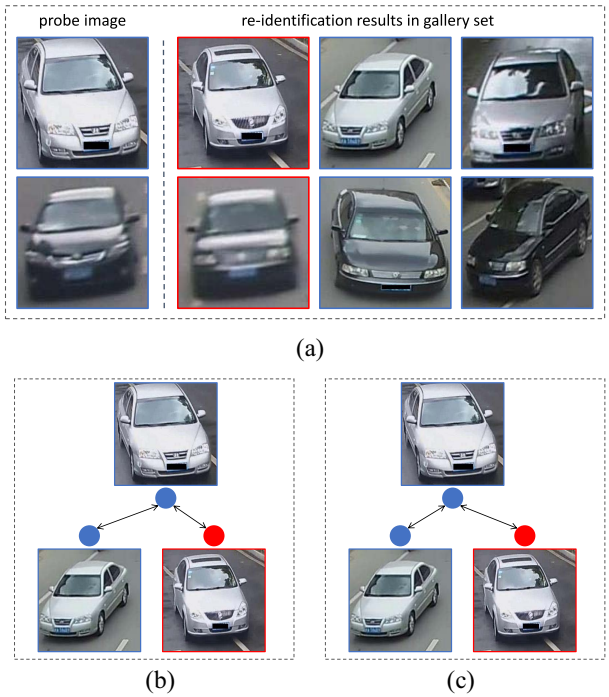

Fig. 1. In each example, the images or features of the same color are of the same identity. (a) Two vehicle reidentification examples. The images on the left are from the probe set, and the images on the right are from the gallery set. Each row is a retrieval example, and the identity of vehicles with the same color frame is the same. The examples show that vehicle pose influences reidentification accuracy. (b) In previous works, two different vehicles are recognized as more similar from a similar viewpoint. (c) In our work, the vehicles of the same identity have a smaller distance.

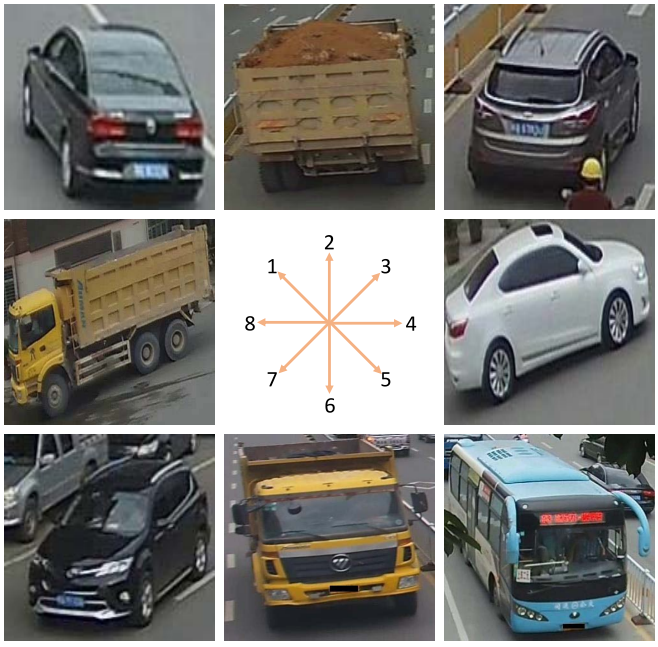

Fig. 2. Illustration of different vehicle orientations. We roughly divided the direction of the vehicle into eight categories: 1) left rear, 2) rear, 3) right rear, 4) right, 5) right front, 6) front, 7) left front, and 8) left.

summation manner. Although this method can exploit additional cues, such as person poses and 3-D scene information, it is not applicable to single-shot matching, which provides no motion or spatial information of a person. In vehicle reidentification, Chu *et al.* [8] proposed a viewpoint-aware metric learning approach to address the pose barrier problem. Zhou and Shao [9] proposed to utilize a vehicle image and the corresponding inferred images of other views to bridge the gap between different vehicle poses. However, few works focus on learning a robust feature representation that is less affected by the pose variations.

To solve the pose barrier problem, we introduce vehicle pose (or called orientation) into our vehicle reidentification framework. Inspired by [10] and [11], the vehicle poses are divided into eight categories according to which faces of the vehicle are visible in this view. The examples of different vehicle poses are shown in Fig. 2. In [10], the method proposes the orientation invariant features by extracting several aligned local features based on the vehicle poses. Then, Khorramshahi *et al.* [11] depended on the orientations and presented a dual-path adaptive attention model, in which the global path captures macroscopic vehicle features, and the part path learns the local details. Note that the above works employ key point estimation networks, which is complicated and computationally expensive. Similar to these works, we propose to introduce vehicle poses to help tackle the vehicle reidentification task. More specifically, we utilize different orientations to represent different poses. In this case, the pose estimation network is concise.

In this article, we aim at training pose-invariant features (PIFs), which are discriminative for vehicle identities and contain little pose cues. PIF is discriminative for vehicle identities and is invariant for different orientations. In other words, no matter what the vehicle poses are, the distance of two extracted features of the same vehicle should be short. To accomplish this goal, we train PIF by jointly optimizing the underlying features as well as two classifiers: 1) the identity classifier and 2) the pose classifier. The identity classifier predicts vehicle identities and the pose classifier discriminates between different orientations. In this network, we adopt a gradient reversal layers (GRLs) to train PIF. However, only using PIF is insufficient for vehicle reidentification. The main reason is that PIF pays more attention to global appearance while neglects local details.

To make the vehicle features more robust, we also introduce PDFs. The network of training PDF has a similar architecture with the network of training PIF. More specifically, the network of training PDF contains one feature extractor and two classifiers. The difference between the PDF network and the PIF network is that the PDF network does not employ the GRL. We merge the orientation discrimination task and the reidentification task to get PDF. Thus, PDF can distinguish the identities as well as the orientations.

Overall, the proposed pose robust features (PRFs) involves two complementary parts, i.e., PIF and PDF. The former attempts to learn a PIF embedding to extract global appearance representation by introducing adversarial loss. The latter focuses on mining local details by simultaneously performing vehicle reidentification and orientation classification. Our experimental results on VeRi-776 and VehicleID demonstrate

that our framework can alleviate the influence of various poses and improve vehicle reidentification accuracy.

In summary, this article makes the following main points.

1) A novel PIF learning approach to solve the vehicle reidentification task by introducing adversarial loss, which is utilized to reduce the pose cues in the trained features.
2) A novel pose discriminative feature (PDF) learning approach to mine local details by simultaneously training an identity classifier and an orientation classifier.
3) A complementary framework that is composed of two kinds of features: a) the PIFs that extract overall global appearance representation and b) the PDFs that focus on local details.

The remainder of this article is organized as follows. Related work is discussed in Section II. PIFs and PDFs are described in Section III. Experimental results are shown and analyzed in Section IV, which is followed by the conclusion in Section VI.

II. Related Works

A. Person Reidentification

One of the most critical research content of person reidentification is feature extraction. We define the features which are not extracted by deep learning as traditional features. Many previous methods [6], [12]–[15] learn feature representations by using global color and texture histograms. Some other reidentification techniques [16]–[18] model person appearance by using local features that are extracted from small subregions in images, such as SIFT [19].

With the development of deep learning techniques [20]–[23], Ahmed *et al.* [24] and Xiao *et al.* [25] used convolutional neural network (CNN) models to specifically find more powerful feature representations. Ahmed *et al.* [24] proposed a deep neural network architecture to yield a final estimate score of whether two input images are of the same person or not. Xiao *et al.* [25] mixed several reidentification data sets together and trained a CNN to recognize person identities.

Research on person reidentification also focuses on finding improved similarity metrics [14], [26]. A good similarity metric is to find a mapping from feature space to a new space so that the same person's features are closer than the pairs from different persons. Zheng *et al.* [14] proposed a model named PRDC that aims to maximize the probability of a pair of a true match. Pedagadi *et al.* [26] proposed the LFDA algorithm to maximize the interclass separability by preserving the multiclass modality. In KISSME [27], a Mahalanobis metric is generated by computing the difference between the intraclass and interclass covariance matrix. As an improvement, Liao *et al.* [28] proposed XQDA to learn a more discriminative distance metric and a low-dimensional subspace simultaneously.

Meanwhile, many approaches are designed to solve the pose barrier problems in person reidentification [6]. Wu *et al.* [6] built a model for human appearance as a function of poses, using training data gathered from a calibrated camera. Then, this pose prior is applied in online reidentification to make matching and identification more robust. Zheng *et al.* [29] proposed to align pedestrians to a standard pose in the person reidentification task. Su *et al.* [30] leveraged the human part cues to alleviate the pose variations and proposed a pose-driven deep convolutional model to learn the feature representations.

However, different from human bodies, vehicles are rigid objects, which make the above approaches are not suitable for vehicle reidentification. The pose barrier problem, which mainly arises from pose variations, is a critical problem for a robust vehicle reidentification system. In this article, we focus on addressing the pose variation problem in vehicle reidentification.

B. Vehicle Reidentification

As several vehicle data sets [1]–[3] are proposed, vehicle reidentification has attracted more attention in the past several years. For example, Liu *et al.* [1], [2] organized the VeRi-776 data set that aims at vehicle reidentification. Based on this work, Wang *et al.* [10] labeled the vehicle orientations and key points. Liu *et al.* proposed the VehicleID data set, which contains vehicles with many nonoverlapping surveillance cameras. The vehicle image in VehicleID is either captured from the front or the back. In this article, we mainly conduct our experiments on VeRi-776 [1], [2] and VehicleID [3].

Recently, many works have attempted to solve the vehicle reidentification problem by using deep models [3], [31], [32]. Liu *et al.* [3] proposed a pipeline and introduced deep relative distance learning (DRDL) to project vehicle images into a Euclidean space. This method can directly measure the similarity of the two compared vehicles. Liu *et al.* [1], [2] built a coarse-to-fine framework by utilizing the visual appearance, license plate, and spatial–temporal information. The VeRi-776 data set has more available points. A two-stage framework is proposed by Shen *et al.* [32]. This method incorporates complex spatiotemporal information for effectively regularizing the reidentification results. Wang *et al.* [10] proposed an orientation invariant feature embedding module and a spatial–temporal regularization module. Chu *et al.* [8] focused on solving the problem caused by extreme viewpoint variation. Chu *et al.* [8] proposed a viewpoint-aware metric learning approach that learns two metrics for similar viewpoints and different viewpoints in two feature spaces.

C. Generative Adversarial Nets

Generative adversarial nets (GANs) [33] is proposed by Goodfellow *et al.*, and this work aims to build deep generative models that can synthesize samples. Recently, to enhance the invariance of inputs, there has been a growing interest in using GAN to augment training data. Zheng *et al.* [34] first introduced the use of unconditional GAN to generate unlabeled images from random vectors. Then, the authors propose the label smoothing regularization for outliers, which assigns a uniform label distribution to the unlabeled images. Moreover, GAN-based image generation methods [35]–[38] are beneficial for reidentification. These methods make use of pose estimation to conduct pose-conditioned image generation. Ma *et al.* [35] first proposed a two-stage image generation method based on the pose. Siarohin *et al.* [36] processed image

generation by using affine transformation modules and the nearest neighbor loss. Li *et al.* [37] proposed to estimate dense and intrinsic 3-D appearance flow and thus better guided the transfer of pixels between poses.

In addition, some works utilize GAN-based approaches to bride the gap between different data distributions. To alleviate the image style variations caused by different cameras, Zhong *et al.* [39] learned image–image translation models for each pair with CycleGAN [40]. As person reidentification suffers from distribution variations between different scenarios, Deng *et al.* [41] utilized CycleGAN [40] to address the problem that deep models trained on one domain often fail to generalize well to another. Ganin and Lempitsky [42] and Ganin *et al.* [43] adopted an adversarial loss to learn domain-invariant features that have similar data distributions in both source and target domains. In vehicle reidentification, vehicle pose variations cause different data distributions. Some works [38], [44], [45] attempt to utilize GAN for vehicle reidentification due to pose challenges. To improve the feature discriminative capability, Lou *et al.* [44] proposed an end-to-end embedding adversarial learning network that can generate samples localized in the embedding space. Lv *et al.* [38] combined perspective transformation and GAN to synthesis multiview vehicle images, which are beneficial to the vehicle reidentification task. Zhou and Shao [45] transformed vehicle features of one input image into a global multiview representation, which makes pairwise distance metric learning better optimized. In this article, we propose the PIFs that have similar data distributions for vehicles of different orientations.

III. Pose Robust Features for Vehicle Reidentification

In this article, we propose PRFs that combine PIFs and PDFs. In this section, we first formulate the vehicle reidentification problem in Section III-A. Then, we describe the baseline of our method in Section III-B. The details of PDF and PIF are described in Sections III-C and III-D, respectively. Finally, the combination of the two features is shown in Section III-E.

A. Problem Formulation

The overall target of vehicle reidentification is to find the queried vehicle from nonoverlapping cameras. When given a query vehicle image, we first get a ranking list of candidates in the gallery set. Then, the vehicles in top positions are taken out as the same vehicle as to the query image.

The problem can be formulated as follows. In this set, we have N images with M different vehicles. $(x_i, m_i, n_i)_{i=1}^{N}$ denotes all training samples, where $x_i \in \{1, 2, \ldots, N\}$ is the ith image. $m_i \in \{1, 2, \ldots, M\}$ is the identity of the corresponding vehicle and $n_i \in \{1, 2, \ldots, 8\}$ is its orientation. Given that all the images with eight orientations, we use the ID-discriminative embedding (IDE) [46] to train the base vehicle reidentification model. The IDE makes reidentification as an image classification where each image contains one ID by using the Softmax loss. Most of the networks, e.g., ResNet [21], Google Inception [47], and DenseNet [48], can be utilized as the backbone of PDN. Following the backbone, several hidden fully connected layers are deployed to fulfill the multitask network. The base backbone is pretrained on ImageNet [49].

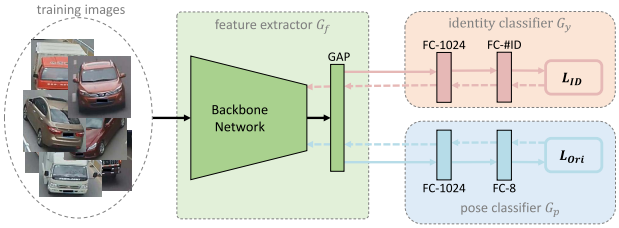

Fig. 3. Network of training PDFs. The network mainly contains three components: the feature extractor G_f, the identity classifier G_y, and the pose classifier G_p.

B. Vechile Reidentification Baseline

Backbone Network: The baseline takes part of DenseNet-121 [48] as the backbone. Before training for vehicle reidentification, the backbone is pretrained on ImageNet [49]. This network can also apply some other backbones, e.g., ResNet [21] and Google Inception [47]. In this article, we deploy the DenseNet-121 [48] as the default backbone, which is proved to be the most effective in our experiments.

Input Size: Since the shape of most vehicle images is similar to a square, we resize each image into 384×384 pixels. Then, we pad the resized image 10 pixels with zero values. Meanwhile, we also resize vehicle images into 128×128, 256×256, and 512×512 pixels to evaluate the influence caused by different input sizes. We can draw a conclusion by the experiments in Section IV. The conclusion is that a larger input size can lead to a higher mean average precision (mAP) in vehicle reidentification. However, when expanding the height and width, more computing time and resources are required. Thus, in this article, we mainly resize the images into 384×384 pixels.

Random Erasing Augmentation (REA): In the baseline, we adopt REA [50]. REA is a data augmentation method and is utilized in person reidentification. In [50], REA improves the generalization ability of deep models and addresses the occlusion problem. The main motivation of REA in vehicle reidentification is to generate images with various levels of occlusion. In training, we conduct REA with a certain probability. For a vehicle image I, we have a probability of p to adopt random erasing, and the probability of it being kept unchanged is $1 - p$. In this process, training images with various levels of occlusion are generated. Finally, we feed the processed images into the vehicle reidentification model.

C. Pose Discriminative Features

PDFs are designed to extract vehicle features that contain pose cues. As shown in Fig. 3, the network of training PDF consists of a feature extractor and two classifiers. The two classifiers are identity classifier G_y and pose classifier G_p, respectively. For each classifier, two fully convolutional layers, as well as a loss function, are adopted. Given an input image,

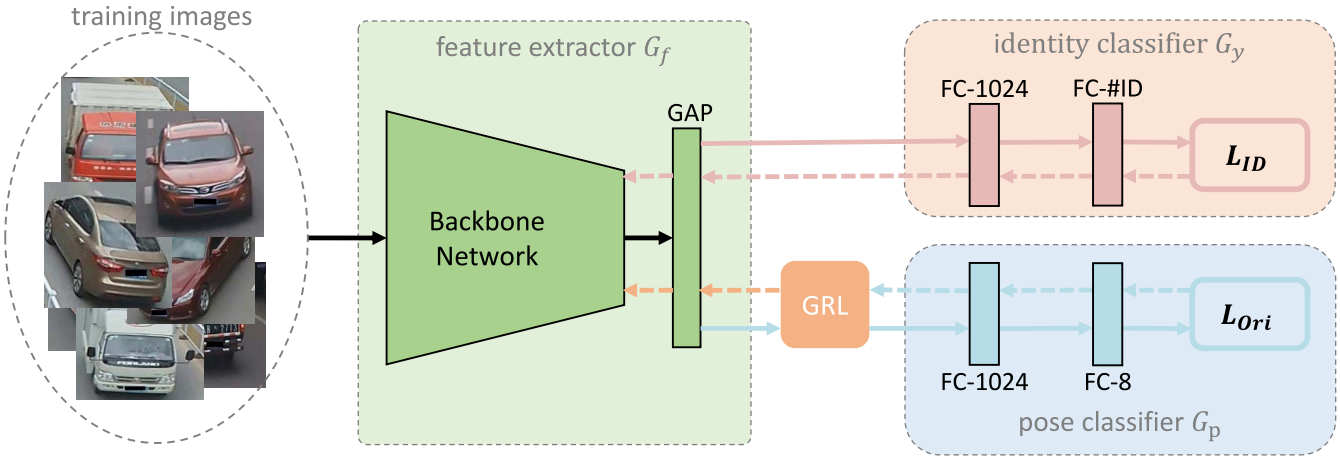

Fig. 4. Network of training PIFs. This network also contains to three components but deploys a GRL between the deep feature extractor (G_f) and the orientation classifier (G_p). During the forward propagation period, GRL does nothing to the input vector and delivers it directly to the following layer. During the backward propagation period, GRL multiplies the gradient by a certain negative constant. This gradient reversal operation makes the feature distributions of G_f over the different orientations similar (as indistinguishable as possible for the pose classifier), thus resulting in the PIFs.

the network can simultaneously yield the corresponding identity as well as the orientation. As shown in Fig. 3, given a training image x_i, the network computes its feature maps by the global average pooling (GAP) layer and outputs a vector. The size of the GAP vector is 256. Note that we adopt the GAP vector as the vehicle feature representation in the testing phase. Subsequently, the network predicts the vehicle orientation and identity based on the image GAP vector. The orientation loss is calculated by the orientation prediction and the orientation class. Similarly, the identity loss is obtained by the identity prediction and the ground-truth identity label. There are two branches for the network: 1) predicting the identity and 2) estimating the orientation.

The first branch is the identity classifier G_y that designed for identity prediction. G_y is composed of an FC-1024, an FC-#ID, and a loss function. FC-1024 means that the length of the fully convolutional layer is 1024 and #ID is the number (M) of people existing in the training set. In this part, we adopt the cross-entropy loss to train the identity branch and the orientation branch. Cross-entropy loss is generally utilized to measure the performance of a classification model whose output is a probability value between 0 and 1. Cross-entropy loss increases as the predicted probability diverges from the ground-truth label. The predicted probability of each ID label $k(\in 1, 2, \ldots, M)$ is calculated as: $p(k|x) = ([\exp(p_k)]/[\sum_{i=1}^{M} \exp(p_i)])$. Thus, the final cross-entropy loss of the identity task can be written as

$$\mathcal{L}_{\text{ID}} = -\sum_{k=1}^{M} \log(p(k))q(k) \tag{1}$$

where $q(k) = 1$ if k equals the identity of the input image and $q(k) = 0$ for all the rest.

The second branch is the pose classifier G_p, which is to estimate the vehicle orientations. It also has two fully convolutional layers. One is FC-1024 and the other one is FC-8. There are eight classes for the orientation task. The probability of assigning input x to the orientation $j \in (1, 2, \ldots, 8)$ can be written as $p(j|x) = ([\exp(p_j)]/[\sum_{i=1}^{8} \exp(p_i)])$. Similarly, we define the loss of the second task as

$$\mathcal{L}_{\text{ori}} = -\sum_{j=1}^{8} \log(p(j))q(j) \tag{2}$$

where $q(j) = 1$ if the j is the ground-truth orientation label.

Then, we combine $\mathcal{L}_{\text{ID}}$ and $\mathcal{L}_{\text{ori}}$ to obtain the final loss function

$$\mathcal{L} = \mathcal{L}_{\text{ID}} + w \cdot \mathcal{L}_{\text{ori}} \tag{3}$$

where w controls the relative importance of the losses.

D. Pose-Invariant Features

In this part, we focus on learning PIFs, which are: 1) discriminative for vehicles with different identities and 2) invariant for different orientations. We achieve PIF by jointly optimizing the underlying features as well as two classifiers: 1) the identity classifier G_y that predicts vehicle identities and 2) the pose classifier G_p that discriminates between different poses. As shown in Fig. 4, when training PIF, the system can be divided into three parts: the two classifiers (G_y and G_p) and the feature extractor G_f.

Motivation: In this part, we focus on leaning pose-invariant representations by combining deep feature learning and pose classification within one training phase. Most previous works train fixed features, which contain pose cues. For example, for two images of a specified vehicle with different orientations, the fixed features have different distributions. In this case, the vehicles with similar orientations are more easily identified as the same vehicle. The main reason is that the distributions of the features are determined by identities as well as poses. However, an optimal feature embedding should be invariant for poses and have similar distributions regardless of different poses. Goodfellow *et al.* [33] simultaneously trained two models: 1) a generative model G and 2) a discriminative model D. To learn domain-invariant features that contain little domain

cues, Deng *et al.* [41] introduced adversarial learning by generating images with different domain styles. Inspired by the above works, we propose PIF that contains few orientation cues and cannot discriminate between the different poses. The difference between our work and [41] is that [41] performs adversarial learning on an image level while PIF on the feature level. Comparing with [41], the proposed method has a much more concise network structure and saves computing resources.

In addition, some machine learning algorithms, i.e., K-nearest neighbor salience [16] and affine CNNs [29], can be adopted to solve the pose barrier problem. In [16], KNN distance is applied to person reidentification to search for the K-nearest neighbors of a test patch in the output set of a dense correspondence. In [29], affine projection is deployed in a CNN structure to effectively alleviate the misalignment problem. Although the above works [16], [29] can improve identification accuracy, they have several defects in solving the pose barrier problem. For example, Zhao *et al.* [16] focused on extracting local details but cannot help output global PIFs. To utilize affine CNNs, the method in [29] is based on a pose estimation model, which is complicated and time consuming. Comparing to the above works [16], [29], utilizing GAN to solve the pose barrier problem is concise and can train PIFs.

We now define the feedforward architecture of extracting PIF. For each input x_i, as shown in Fig. 4, the proposed network predicts the identity y_i and the pose p_i. We decompose the overall architecture into three parts: 1) the feature extractor G_f; 2) the identity classifier G_y; and 3) the pose classifier G_p. The feature extractor include an existing deep backbone model, i.e., DenseNet [48], and several feedforward layers. The parameters of the feature extractor is defined as θ_f and the extracted feature is defined as $f_i = G_f(x_i; \theta_f)$. Note that f is utilized to calculate the distance between vehicles during the testing phase. Following the feature extractor, the identity classifier G_y maps f to the label y and the pose classifier G_p maps f to pose p. The parameters of the identity classifier and the pose classifier are θ_y and θ_p, respectively.

The training process includes two main parts. On the one hand, we intend to ensure the discriminativeness of f when distinguishing vehicle identity. This discriminativeness is achieved by minimizing identity prediction loss. During the training stage, the identity loss $\mathcal{L}_{\mathrm{ID}}$ is fed into the network and the parameters of both the feature extractor G_f and the identity classifier G_y are thus optimized. On the other hand, we want to make the features f pose invariant. In other words, f should contain as few pose details as possible and the distributions of the vehicles with different poses should be similar. In conclusion, in order to obtain PIF, we propose to train the parameters θ_f of the feature f that maximize the loss of the pose classifier G_p and minimize the loss of identity classifier G_y. More formally, the functional can be written as

$$\begin{aligned} E(\theta_f, \theta_y, \theta_p) &= \sum_{i=1}^{M} \mathcal{L}_y^i(\theta_f, \theta_y) + w \cdot (-\lambda) \sum_{i=1}^{M} \mathcal{L}_p^i(\theta_f, \theta_p) \\ &= \sum_{i=1}^{M} \mathcal{L}_y^i(\theta_f, \theta_y) - w \cdot \lambda \sum_{i=1}^{M} \mathcal{L}_p^i(\theta_f, \theta_p) \end{aligned} \tag{4}$$

where $\mathcal{L}_y^i$ denotes the loss for identity classifier G_y, which is calculated on the ith sample. Similarly, $\mathcal{L}_p^i$ is the loss for the pose classifier. The parameter w controls the relative importance of the losses. $-\lambda$ provides the reversal gradient between the feature extractor G_f and the pose classifier G_p. In order to train PIF f, we train the parameters $\hat{\theta}_f$, $\hat{\theta}_y$, and $\hat{\theta}_p$ to satisfy the following:

$$\left(\hat{\theta}_f, \hat{\theta}_y\right) = \arg\min_{\theta_f, \theta_y} E\left(\theta_f, \theta_y, \hat{\theta}_p\right) \tag{5}$$

$$\hat{\theta}_p = \arg\max_{\theta_p} E\left(\hat{\theta}_f, \hat{\theta}_y, \theta_p\right). \tag{6}$$

In our method, the parameters θ_p minimize the pose classification loss and the parameters θ_y minimize the identity classification loss. Meanwhile, the parameters θ_f of the feature extractor G_f serve two purposes: the parameters θ_f minimize the identity prediction while maximizing the pose classification loss. Note that the deployment of $-\lambda$ achieves the process of maximizing the pose classification loss.

Training PIF: The parameters θ_f, θ_y, and θ_p are updated with backward propagation. The update process of the parameters can be described as

$$\theta_f \leftarrow \theta_f - \mu\left(\frac{\partial \mathcal{L}_y^i}{\partial \theta_f} - w \cdot \lambda \frac{\partial \mathcal{L}_p^i}{\partial \theta_f}\right) \tag{7}$$

$$\theta_y \leftarrow \theta_y - \mu \frac{\partial \mathcal{L}_y^i}{\partial \theta_y} \tag{8}$$

$$\theta_p \leftarrow \theta_p - w \cdot \mu \frac{\partial \mathcal{L}_p^i}{\partial \theta_p} \tag{9}$$

where μ is the learning rate in the reidentification system. The update procedure is similar to the stochastic gradient descent (SGD) and the difference is the parameter λ. In updating θ_f, we use $-\lambda$ to make features f similar across different poses by maximizing the pose classification loss $\partial \mathcal{L}_p^i / \partial \theta_f$.

GRL: In order to accomplish the above training process, i.e., the implementation of $-\lambda$, we introduce GRL, which is located between the feature extractor G_f and the pose classifier G_p. GRL has the following features: 1) GRL contains a meta-parameter λ and has no parameters; 2) during the forward propagation, GRL has no processing of forward parameters and transfers the weights to the subsequent layers; and 3) during the backward propagation, GRL multiplies the gradient by $-\lambda$ and passes it to the preceding layer. Note that the gradient is taken from the subsequent layer. GRL performs in two phases: 1) the forward propagation phase and 2) the backward propagation phase. *In the forward propagation phase,* GRL only transfers input vectors to the following layers without performing other operations. More formally, let x be the input vector and R_λ be GRL. The forward propagation is defined as

$$R_\lambda(x) = x. \tag{10}$$

In the backward propagation phase, GRL takes the gradient from the subsequent level (pose classifier G_p) and multiplies it by $-\lambda$. The processed gradients then are passed to the previous layer (feature extractor G_f). The backward

propagation function of GRL can be written as

$$\frac{dR_\lambda}{dx} = -\lambda I. \tag{11}$$

We gradually change λ from 0 to 1 on the basis of the number of network updates p and a parameter γ. λ is calculated as

$$\lambda = \frac{2}{1 + e^{-\gamma p}} - 1. \tag{12}$$

Extracting Features: After the training process is finished, we utilize the feature extractor G_f to generate vehicle representations, which are GAP vectors. The size of each vector is also 256. As we introduce GRL to train the PIFs, the network structure can be regarded as an adversarial system, which consists of a generator and a discriminator. Specifically, the feature extractor G_f is the generator and the pose classifier G_p is the discriminator. As we aim at training PIF that contains little orientation cues, feature extractor G_f is trained by an adversarial loss to fool the discriminator. Meanwhile, the pose classifier G_p acts as a discriminator that distinguishes the generated features produced by the generator from the true orientations. Similar to other GAN-based methods, we use the features generated by the feature extractor G_f to perform vehicle reidentification.

E. Combining Two Features

As PRFs consists of two features, i.e., PIF and PDF, we obtain PRF by combining PIF and PDF in the framework of vehicle reidentification. Suppose that we have f_I and f_D, which are PIF and PDF, respectively. In order to measure the similarity between two vehicles, the complete form of PRF (f_R) is defined as

$$f_R = \left[f_I, \alpha f_D\right] \tag{13}$$

where α controls the relative importance of the two features. Then, the Euclidean distance (L2) is adopted to compute the similarity score between query and gallery images at the testing stage. The detailed algorithm of obtaining PRF is described in Algorithm 1.

Algorithm 1: How to Obtain the PRFs. PRF Consists of Two Parts: PDFs and PIFs

Input: Set of training samples $(x_i, m_i, n_i)_{i=1}^N$. $x_i \in \{1, 2, \cdots, N\}$ is the ith image, $m_i \in \{1, 2, \cdots, M\}$ is the identity of the corresponding vehicle, and $n_i \in \{1, 2, \cdots, 8\}$ is its orientation.

PDF: (Section III-C) *Training.* Updating the network in Fig. 3 by the backward propagation of the calculated loss:

$$\mathcal{L} = \mathcal{L}_{\text{ID}} + w \cdot \mathcal{L}_{\text{ori}}.$$

Extracting f_D. After the above network is trained, we extract PDF f_D by calculating the feature maps of the GAP layer. The output of the GAP layer is a vector of length 256.

PIF: (Section III-D) *Training.* The parameters of the overall network are divided into three parts: θ_f, θ_y and θ_p. We update the parameters by backward propagation together with the GRL. The update process of the parameters can be described as,

$$\theta_f \leftarrow \theta_f - \mu\left(\frac{\partial \mathcal{L}_y^i}{\partial \theta_f} - w \cdot \lambda \frac{\partial \mathcal{L}_p^i}{\partial \theta_f}\right),$$

$$\theta_y \leftarrow \theta_y - \mu \frac{\partial \mathcal{L}_y^i}{\partial \theta_y},$$

$$\theta_p \leftarrow \theta_p - w \cdot \mu \frac{\partial \mathcal{L}_p^i}{\partial \theta_p}.$$

Extracting f_I. Similar to PDF, we extract PIF f_I by calculating the GAP vector. Note that GRL only plays a role in forward propagation when training the network. Thus, GRL is not utilized during testing phase.

PRF: (Section III-E) After the above process, PRF f_R is obtained by combining PDF f_D and PIF f_I. The complete form of PRF f_R is defined as,

$$f_R = [f_I, \alpha f_D].$$

IV. Experiments

A. Data Sets and Evaluation Protocol

There are mainly four existing vehicle data sets related to vehicle reidentification, including VeRi-776 [1], [2], VehicleID [3], BoxCars21k [51], and CompCars [52]. VeRi-776 [1], [2] and VehicleID [3] are benchmark datesets for vehicle reidentification. BoxCars21k [51] and CompCars [52] focus on fine-grained recognition. In this article, we conduct our experiments on the two large-scale vehicle reidentification benchmarks: 1) VeRi-776 [1], [2] and 2) VehicleID [3].

VeRi-776 [1], [2] is collected from real-world surveillance scenarios, with over 50 000 images of 776 vehicles in total. The images are from 20 cameras, and each vehicle is captured by 2–18 cameras at different viewpoints. This training subset contains 576 vehicles with 37 781 images, and the test subset contains 200 vehicles with 1678 images. The VeRi-776 also contains spatiotemporal details, such as the timestamps of vehicles and the distances between neighboring cameras. However, we focus on solving the problem caused by appearance, and the spatiotemporal details are not used in this article.

VehicleID [3] contains 221 763 images of 26 267 vehicles in total. The images are captured during daytime by multiple real-world surveillance cameras distributed in a small city in China. Similar to VeRi-776, each vehicle ever appeared includes more than one image. Besides, the vehicle in each image is either captured from the front or the back.

Evaluation Protocol: In this experiment, we employ the commonly used cumulative match characteristic (CMC) curves and CMC rank@k accuracy to compare our method with the others. The rank@k matching rate specifies the percentage of probe images that matched correctly with one of the top k images in the gallery set. The CMC curve summarizes the chance of the correct match appearing in the top $1, 2, \ldots, n$

TABLE I
RANK@K (%) AND MAP (%) ACCURACY WITH DIFFERENT BACKBONES ON THE VERI DATA SET. THE BEST RESULTS ARE IN **BOLD**. THE INPUT IMAGES ARE RESIZED TO 256 × 256. NOTE THAT REA IS NOT APPLIED IN THIS PART

Backbones	mAP	rank@1	rank@5	rank@20
Inception	62.34	87.28	92.56	94.61
ResNet-50	64.12	90.05	94.46	95.95
DenseNet-121	**69.69**	**93.80**	**97.14**	**98.09**

TABLE II
RANK@K (%) AND MAP (%) ACCURACY WITH DIFFERENT INPUT SIZES ON THE VERI DATA SET. WE APPLY DENSENET-121 [48] AS THE BACKBONE NETWORK AND USE REA

Input Size	mAP	rank@1	rank@5	rank@20
128 × 128	68.63	92.13	96.60	97.85
256 × 256	72.94	94.05	97.07	98.99
384 × 384	74.41	**94.87**	**98.45**	**99.23**
512 × 512	**74.42**	94.83	97.97	98.93

TABLE III
EFFECT OF REA ON THE VERI-776 DATA SET. THE INPUT IMAGES ARE RESIZED TO 384 × 384. NOTE THAT REA IS NOT APPLIED IN THIS PART

Method	mAP	rank@1	rank@5	rank@20
baseline w/o REA	71.42	94.52	97.44	98.45
baseline	**74.41**	**94.87**	**98.45**	**99.23**

of the ranked list. The first point of the CMC curve is rank@1 accuracy. Note that we only evaluate cross-camera vehicle reidentification. If the probe image and the gallery image are captured under the same camera, the corresponding matching result will be excluded in the final performance evaluation. We also use the mAP proposed by Zheng *et al.* [53] for evaluation.

B. Experiments on Baseline

Backbone Network: The performance of our baseline is highly related to the backbone network. We evaluate the influence of different backbone networks, i.e., Google Inception [47], ResNet-50 [21], and DenseNet-121 [48]. As shown in Table I, the network adopting DenseNet-121 [48] achieves the best performance. Note that the input images are resized to 256 × 256 in this part. The Inception backbone [47] yields 62.34 in mAP and 87.28 in rank@1. By adopting ResNet-50 [21], the network yields an mAP of 64.12 and a rank@1 of 90.05, which are higher than the results of Inception [47]. The above result indicates the effectiveness of the ResNet-50 backbone network. Moreover, compared to the above backbones, DenseNet-121 [48] arrives at the best performance 69.69 mAP and 93.80 rank@1. The results illustrate that the DenseNet-121 backbone network is most suitable for the vehicle reidentification task. Thus, we use DenseNet-121 as the default backbone network in the baseline and the following experiments.

Input Size: We also conduct experiments on different input sizes to explore the impact of resolution on network performance. For comparison, as shown in Table II, the input images have several sizes that range from 128 to 512. On the VeRi-776 data set [1], [2], we deploy the DenseNet-121 [48] as a backbone network and evaluate the influence of different input sizes. When the image size is set to 384 × 384,

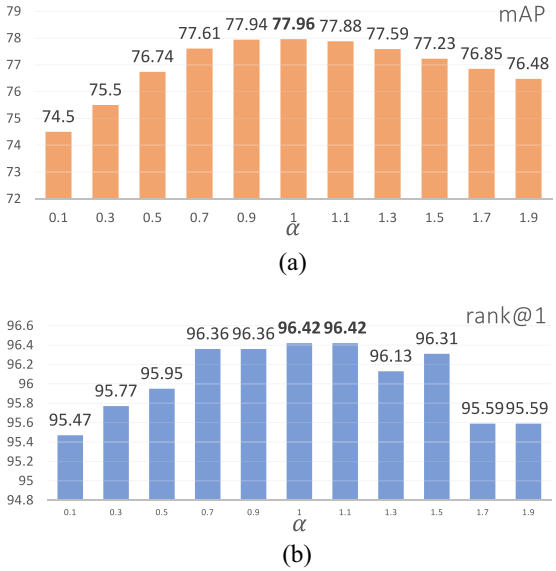

Fig. 5. Evaluation with different α on the VeRi-776 [1], [2] data set. We show the reidentification (a) mAP and (b) rank@1 accuracy, respectively.

the proposed method achieves 74.41 mAP and 94.97 rank@1, which is superior to the set of smaller input sizes, i.e., 128 × 128 and 256 × 256. It shows that larger input images help to get higher mAP and rank@1. When the input size is set to 512 × 512, the mAP is slightly improved from 74.41 to 74.42. Meanwhile, processing larger input images (512 × 512) requires much more computing resources than the smaller images (384 × 384). Thus, in the following experiments, all the input images are set to 384 × 384.

Random Erasing Augmentation: We observe that employing REA [50] leads to better results. As shown in Table III, the baseline arrives at a better performance with 74.41 mAP and 94.87 rank@1 on the VeRi-776 data set. When REA is not adopted in training the models, baseline w/o REA achieves −2.99 mAP and −0.35 rank@1.

C. Implementation Details

Baseline: To learn the baseline model, we adopt IDE+ [59] as the feature learning method. All the images are resized to 384 × 384. During training, we adopt random flipping and random cropping as data augmentation methods. The dropout probability is set to 0.5. We adopt DenseNet-121 [48] as the backbone network. We apply the SGD optimizer to train the models on 1080Ti GPUs in a total of 60 epochs. The weight decay is set to 0.0005. The initial learning rate is set to 0.1 and is divided by a factor 10 at the 40th epoch. We modify the last stride of DenseNet-121 as 1, which leads to larger spatial size and is beneficial for feature learning.

PIF and PDF: To learn PDFs and PIFs, we follow a similar setting as the baseline model. However, different from the baseline model, PIF and PDF have a weight w that controls the relative importance of the losses. In this article, as we aim at obtaining representations that are discriminative for different vehicles, $\mathcal{L}_{\text{ID}}$ has a greater weight than $\mathcal{L}_{\text{ori}}$. In PIF, w is set to 0.1 during the training phase. In PDF, w is set to 1 at the beginning of training and then divided by 10 every 20 epochs. In addition, when training the PIF, γ is set to 10.

TABLE IV
EVALUATION OF DIFFERENT w DURING TRAINING THE PIFS ON VERI-776 [1], [2]

w	mAP	rank@1	rank@5	rank@20
0.1	**74.36**	**94.45**	**97.85**	**98.63**
0.2	72.83	94.44	97.79	98.61
0.3	50.67	84.15	93.98	96.13
0.4	48.24	81.64	91.48	95.11
0.5	42.34	76.70	90.17	93.98

D. Parameter Analysis

An important hyperparameter of PRFs is α described in Section III-E. This parameter is used to control the relative importance of PIF and PDF. We show its impact by varying its value in Fig. 5. We observe that the methods with different α have different results. The best performance is achieved when $\alpha = 1$, which indicates that PIF and PDF are equally important for the overall framework. Note that we set $\alpha = 1$ in the following experiments.

We also discuss the influence of different values for w during the training phase of PIF (Section III-D). w controls the relative importance of identity loss and orientation loss. As shown in Table IV, when w is set to 0.1, PIF achieves an mAP of 74.36, while mAP of the competing settings obtains 72.83, 50.67, 48.24, and 42.34. We also observe that mAP and rank/@k have lower reidentification accuracy as w increases. The reason is that the increase in the weight of orientation loss leads to a decline in the learning ability of identity task.

E. Comparison With the State-of-the-Art Methods

The proposed method is compared with several state-of-the-art vehicle reidentification approaches, i.e., PAMTRI [58], AAVER [11], and VAMI+STR [45]. Performance evaluation is conducted on VeRi-776 [1], [2] and VehicleID [3]. On VeRi-776, we adopt mAP, rank@1, and rank@5 for evaluation. Meanwhile, on VehicleID, we apply rank@1, rank@5, and rank@20 to evaluate the performance of the proposed method.

Evaluation on VeRi-776 [1], [2]: The compared approaches include FACT+STR [1], RAM [55], GSTE [57], S+LSTM [32], VAMI+STR [45], VANet [8], AAVER [11], and PAMTRI [58], where +ST and +STR indicate that spatiotemporal details are utilized in the corresponding approaches. Note that only the vehicle appearance information is involved in our method. As shown in Table V, the proposed method outperforms the competing methods, including those involving additional spatiotemporal details. The proposed method obtains an mAP of 75.43, while mAP of the competing methods is 27.77, 61.50, 59.40, 58.27, 61.32, 66.34, 66.35, and 71.88, respectively. In addition, we also compare rank@1 and rank@5. In terms of rank@1, the proposed method yields a rank@1 of 95.63 on the testing set. On this metric, the proposed method outperforms most of the compared methods.

Evaluation on VehicleID [3]: The compared state-of-the-art methods on VehicleID include DRDL [3], XVGAN [9], CLVR [54], RAM [55], ABLN [56], and GSTE [57]. Table VI shows the comparison results on VehicleID [3]. Our method significantly outperforms the competing methods, including the methods that involve additional labeling information.

TABLE V
EXPERIMENTAL RESULTS OF THE PROPOSED METHOD AND OTHER COMPARED METHODS ON THE VERI-776 [1], [2] DATA SET. MULTIPLE EVALUATION CRITERIA ARE ADOPTED, INCLUDING MAP, RANK@1, AND RANK@5

Methods	mAP	rank@1	rank@5
FACT+STR [1]	27.77	61.44	78.78
RAM [55]	61.50	88.60	94.00
GSTE [57]	59.40	96.24	**98.97**
S+LSTM [32]	58.27	83.49	90.04
VAMI+STR [45]	61.32	85.92	91.84
VANet [8]	66.34	89.78	95.99
AAVER [11]	66.35	90.17	94.34
PAMTRI [58]	71.88	92.86	96.97
Ours	**77.96**	**96.42**	98.51

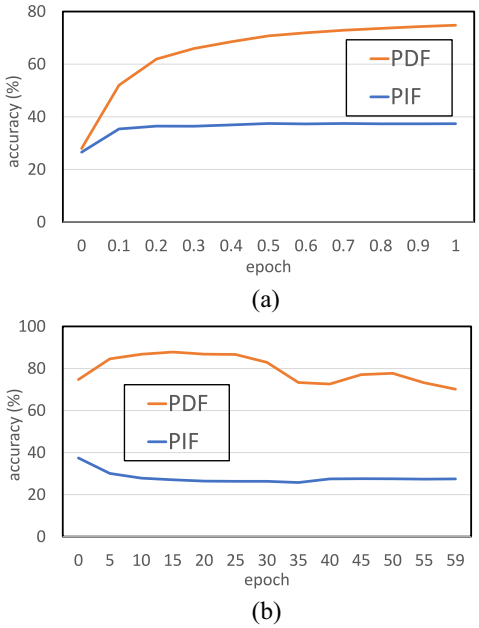

Fig. 6. Curves of the pose classification accuracy when adopting different features on VeRi-776 [1], [2]. We show the pose classification accuracy (a) in the first epoch and (b) throughout the training.

F. Ablation Study

In this section, we present ablation studies of our method. Since two components are involved, i.e., PIF and PDF, we remove them one at a time to evaluate their contributions, respectively. Results on the VeRi-776 data set [1], [2] are shown in Table VII.

Effect of PIF: We first evaluate the PIF, which is designed to extract overall appearance features. As shown in Table VII, on the VeRi-776 [1], [2] data set, PIF yields an mAP of 74.36 and a rank@1 of 94.45, while the mAP and rank@1 of the baseline are 74.41 and 94.87, respectively. It indicates that PIF does not perform as well as the baseline. The main reason is that the baseline focuses on both the global appearance and the local details. However, PIF is only trained to learn a global feature embedding.

Effect of PDF: Similar to PIF, PDF outputs lower mAP (73.58) and rank@1 (94.76). The main reason is that PDF is

TABLE VI
EXPERIMENTAL RESULTS OF THE PROPOSED METHOD AND OTHER COMPARED METHODS ON THE VEHICLEID [3] DATA SET. MULTIPLE EVALUATION CRITERIA ARE ADOPTED, INCLUDING RANK@1, RANK@5, AND RANK@20

Methods	Small			Medium			Large		
	rank@1	rank@5	rank@20	rank@1	rank@5	rank@20	rank@1	rank@5	rank@20
DRDL [3]	48.93	75.65	88.47	45.05	68.85	79.88	41.05	63.38	76.62
XVGAN [9]	52.87	80.83	91.86	49.55	71.39	81.73	44.89	66.65	78.04
CLVR [54]	62.00	76.00	-	56.10	71.80	-	50.60	68.00	-
RAM [55]	75.20	91.50	-	72.30	87.00	-	67.70	84.50	-
ABLN [56]	52.63	80.51	91.25	-	-	-	-	-	-
GSTE [57]	87.10	-	-	82.10	-	-	79.80	-	-
Ours	**88.50**	**97.50**	**99.25**	**83.37**	**95.56**	**98.87**	**80.83**	**93.33**	**98.33**

TABLE VII
ABLATION STUDY OF PRFS ON VERI-776 [1], [2]

Method	mAP	rank@1	rank@5	rank@20
baseline	74.41	94.87	98.45	99.23
PIF	74.36	94.45	97.85	98.63
PDF	73.58	94.76	98.33	99.34
PRF	**77.96**	**96.42**	**98.51**	**99.28**

expected to mine local details and is not good at extract global features.

Effect of PRF: The full PRF system combines the two features and yields an mAP of 77.96. When compared with the baseline approach, employing PIF or PDF alone brings +3.55, +3.50, and 4.38 improvement in mAP. Comparing to the baseline approach, the experiments show that PRF is effective and competitive. Comparing to PIF and PDF, the combination clearly indicates that the two features are complementary.

PDF Contains Pose Cues While PIF Contains Few: To evaluate whether PDF and PIF contain pose cues, we introduce pose classification accuracy during the training phase of PDF and PIF. In training PDF, as shown in Fig. 6(a), the accuracy rises quickly, reaching 74.79 in the first epoch. At the beginning of training, PDF focuses on learning discriminative identity features as well as pose details. As the training progresses, as shown in Fig. 6(b), the accuracy rate is up to 87, which indicates that the pose classifier is not sufficiently trained. The main reason for the inadequate training is that the orientation labels of VeRi-776 [1], [2] are not accurate enough. For example, when the orientation of a vehicle is between the front and right front, this orientation can be defined as the front or right front. After arriving at the maximum value, the classification accuracy of PDF begins to decline, which is due to the decrease in the value of loss weight w. However, the classification accuracy of PIF is low and keeps about 25% throughout the training phase. It clearly indicates that PIF contains few pose details, which makes PIF invariant for poses.

V. DISCUSSION

Open Issues: With the advancement of deep learning technologies and increasing demand for intelligent video surveillance, vehicle reidentification has attracted significant attention in the computer vision community. In the IoT community, vehicle reidentification can contribute to smart city applications, including traffic monitoring and vehicle retrieval. However, there are still a couple of open issues that need to be addressed in future work. *On the one hand*, one issue is how to calculate the feature distance between vehicles that have various orientations. During training, we deploy the pose classification task to learn PDFs and PIFs. During testing, we only use the Euclidean distance to compute the similarity score between query and gallery images. In other words, the challenges caused by various poses are not considered in the inference stage. *On the other hand*, to apply algorithms to IoT devices, processing speed is also an issue that needs to be considered. Although the proposed method is simple and effective, the speed of extracting vehicle features still has room for improvement.

Future Research Plan: To address the first issue, a pose-aware metric learning approach could be developed in the inference stage. A distance metric learning approach can automatically construct distance metrics from labeled vehicle images in a machine learning manner. In this article, we simply combine PDF and PIF and then calculate the Euclidean distance between vehicles. Different from the above inference method, we aim at proposing a pose-aware metric learning approach that is based on PDF and PIF. The metric learning approach should calculate the distance in a more reasonable way by providing different weights for features of different poses. For the second issue, we plan to compress deep models with pruning, trained quantization, or the Huffman coding based on the proposed PRF framework.

VI. CONCLUSION

In this article, we proposed to learn PRFs to address the pose barrier problem in vehicle reidentification. The overall framework consisted of two main components, i.e., PIFs and PDFs. To train PDF, we built a two-branch deep model by simultaneously training an identity classifier and a pose classifier. The trained PDF focused on local details and was discriminative for different orientations. To train PIF, we deployed a GRL between the feature extractor and the pose classifier to enforce the learned features invariant for poses. To verify the effectiveness of PRF, we evaluated the proposed method on VeRi-776 and VehicleID. On the one hand, we showed that PRF, which combines PIF and PDF and achieves state-of-the-art results. On the other hand, the ablation study illustrated PIF and PDF were complementary.

ACKNOWLEDGMENT

The authors would like thank the HAWKEYE Group for the support.

REFERENCES

[1] X. Liu, W. Liu, T. Mei, and H. Ma, "A deep learning-based approach to progressive vehicle re-identification for urban surveillance," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 869–884.

[2] X. Liu, W. Liu, H. Ma, and H. Fu, "Large-scale vehicle re-identification in urban surveillance videos," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2016, pp. 1–6.

[3] H. Liu, Y. Tian, Y. Yang, L. Pang, and T. Huang, "Deep relative distance learning: Tell the difference between similar vehicles," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2167–2175.

[4] Y. Cho and K. Yoon, "Improving person re-identification via pose-aware multi-shot matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1354–1362.

[5] S. Bak, F. Martins, and F. Bremond, "Person re-identification by pose priors," in *Proc. Int. Soc. Opt. Eng.*, vol. 9399, 2015, Art. no. 93990H.

[6] Z. Wu, Y. Li, and R. Radke, "Viewpoint invariant human re-identification in camera networks using pose priors and subject-discriminative features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 5, pp. 1095–1108, May 2015.

[7] H. Sheng *et al.*, "Mining hard samples globally and efficiently for person re-identification," *IEEE Internet Things J.*, early access, Mar. 13, 2020, doi: 10.1109/JIOT.2020.2980549.

[8] R. Chu, Y. Sun, Y. Li, Z. Liu, C. Zhang, and Y. Wei, "Vehicle re-identification with viewpoint-aware metric learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 8282–8291.

[9] Y. Zhou and L. Shao, "Cross-view GAN based vehicle generation for re-identification," in *Proc. Brit. Mach. Vis. Conf.*, vol. 1, 2017, pp. 1–12.

[10] Z. Wang *et al.*, "Orientation invariant feature embedding and spatial temporal regularization for vehicle re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 379–387.

[11] P. Khorramshahi, A. Kumar, N. Peri, S. S. Rambhatla, J. Chen, and R. Chellappa, "A dual-path model with adaptive attention for vehicle re-identification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 6132–6141.

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

[13] H. Sheng *et al.*, "Hypothesis testing based tracking with spatio-temporal joint interaction modeling," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Apr. 20, 2020, doi: 10.1109/TCSVT.2020.2988649.

[14] W. Zheng, S. Gong, and T. Xiang, "Person re-identification by probabilistic relative distance comparison," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 649–656.

[15] C. Engel, P. Baumgartner, M. Holzmann, and J. F. Nutzel, "Person re-identification by support vector ranking." in *Proc. Brit. Mach. Vis. Conf.*, 2010, pp. 1–11.

[16] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3586–3593.

[17] Y. Zhang *et al.*, "Long-term tracking with deep tracklet association," *IEEE Trans. Image Process.*, vol. 29, pp. 6694–6706, May 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9096592

[18] R. Zhao, W. Ouyang, and X. Wang, "Learning mid-level filters for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 144–151.

[19] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[20] C. Szegedy, W. Liu, Y. Jia, and P. Sermanet, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.

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

[22] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A survey on Internet of Things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1125–1142, Oct. 2017.

[23] W. G. Hatcher and W. Yu, "A survey of deep learning: Platforms, applications and emerging research trends," *IEEE Access*, vol. 6, pp. 24411–24432, 2018.

[24] E. Ahmed, M. Jones, and T. Marks, "An improved deep learning architecture for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3908–3916.

[25] T. Xiao, H. Li, W. Ouyang, and X. Wang, "Learning deep feature representations with domain guided dropout for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1249–1258.

[26] S. Pedagadi, J. Orwell, S. Velastin, and B. Boghossian, "Local fisher discriminant analysis for pedestrian re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3318–3325.

[27] P. M. Roth, P. Wohlhart, M. Hirzer, M. Kostinger, and H. Bischof, "Large scale metric learning from equivalence constraints," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2288–2295.

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

[29] L. Zheng, Y. Huang, H. Lu, and Y. Yang, "Pose-invariant embedding for deep person re-identification," *IEEE Trans. Image Process.*, vol. 28, no. 9, pp. 4500–4509, Sep. 2019.

[30] C. Su, J. Li, S. Zhang, J. Xing, W. Gao, and Q. Tian, "Pose-driven deep convolutional model for person re-identification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3960–3969.

[31] K. Lv *et al.*, "Vehicle re-identification with location and time stamps." in *Proc. CVPR Workshops*, 2019, pp. 399–406.

[32] Y. Shen, T. Xiao, H. Li, S. Yi, and X. Wang, "Learning deep neural networks for vehicle re-id with visual-spatio-temporal path proposals," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1918–1927.

[33] I. J. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[34] Z. Zheng, L. Zheng, and Y. Yang, "Unlabeled samples generated by GAN improve the person re-identification baseline in vitro," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3754–3762.

[35] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool, "Pose guided person image generation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 405–415.

[36] A. Siarohin, E. Sangineto, S. Lathuilière, and N. Sebe, "Deformable GANs for pose-based human image generation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3408–3416.

[37] Y. Li, C. Huang, and C. C. Loy, "Dense intrinsic appearance flow for human pose transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3693–3702.

[38] K. Lv, H. Sheng, Z. Xiong, W. Li, and L. Zheng, "Pose-based view synthesis for vehicles: A perspective aware method," *IEEE Trans. Image Process.*, vol. 29, pp. 5163–5174, Mar. 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9042874

[39] Z. Zhong, L. Zheng, Z. Zheng, S. Li, and Y. Yang, "Camera style adaptation for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5157–5166.

[40] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2223–2232.

[41] W. Deng, L. Zheng, Q. Ye, G. Kang, Y. Yang, and J. Jiao, "Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 994–1003.

[42] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," 2014. [Online]. Available: arXiv:1409.7495.

[43] Y. Ganin *et al.*, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2030–2096, 2016.

[44] Y. Lou, Y. Bai, J. Liu, S. Wang, and L.-Y. Duan, "Embedding adversarial learning for vehicle re-identification," *IEEE Trans. Image Process.*, vol. 28, no. 8, pp. 3794–3807, Aug. 2019.

[45] Y. Zhou and L. Shao, "Viewpoint-aware attentive multi-view inference for vehicle re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6489–6498.

[46] L. Zheng, Y. Yang, and A. G. Hauptmann, "Person re-identification: Past, present and future," 2016. [Online]. Available: arXiv:1610.02984.

[47] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-V4, inception-RESNET and the impact of residual connections on learning." in *Proc. AAAI*, vol. 4, 2017, p. 12.

[48] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4700–4708.

[49] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[50] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," 2017. [Online]. Available: arXiv:1708.04896.

[51] J. Sochor, A. Herout, and J. Havel, "BoxCars: 3D boxes as CNN input for improved fine-grained vehicle recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3006–3015.

[52] L. Yang, P. Luo, C. Change Loy, and X. Tang, "A large-scale car dataset for fine-grained categorization and verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3973–3981.
[53] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1116–1124.
[54] A. Kanacı, X. Zhu, and S. Gong, "Vehicle reidentification by fine-grained cross-level deep learning," in *Proc. Brit. Mach. Vis. Conf. AMMDS Workshop*, vol. 2, 2017, pp. 772–788.
[55] X. Liu, S. Zhang, Q. Huang, and W. Gao, "RAM: A region-aware deep model for vehicle re-identification," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2018, pp. 1–6.
[56] Y. Zhou and L. Shao, "Vehicle re-identification by adversarial bi-directional LSTM network," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2018, pp. 653–662.
[57] Y. Bai, Y. Lou, F. Gao, S. Wang, Y. Wu, and L.-Y. Duan, "Group-sensitive triplet embedding for vehicle reidentification," *IEEE Trans. Multimedia*, vol. 20, no. 9, pp. 2385–2399, Sep. 2018.
[58] Z. Tang *et al.*, "PAMTRI: Pose-aware multi-task learning for vehicle re-identification using highly randomized synthetic data," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 211–220.
[59] L. Zheng *et al.*, "MARS: A video benchmark for large-scale person re-identification," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 868–884.

Hao Sheng (Member, IEEE) received the B.S. and Ph.D. degrees from the School of Computer Science and Engineering, Beihang University, Beijing, China, in 2003 and 2009, respectively.

He is currently an Associate Professor with the School of Computer Science and Engineering, Beihang University. He is working on computer vision, pattern recognition, and machine learning.

Kai Lv received the B.S. degree from the School of Computer Science and Technology, Tianjin University of Science and Technology, Tianjin, China, in 2012. He is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, Beihang University, Beijing, China.

Yang Liu received the B.S. degree from the School of Advanced Engineering, Beihang University, Beijing, China, in 2009, where he is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering.

His research interests include deep learning, computer vision, and especially person reidentification.

Wei Ke received the Ph.D. degree from the School of Computer Science and Engineering, Beihang University, Beijing, China, in 2012.

He is an Associate Professor of computing program with Macao Polytechnic Institute, Macau, China. His research interests include programming languages, image processing, computer graphics, and tool support for object-oriented and component-based engineering and systems. His recent research focuses on the design and implementation of open platforms for applications of computer graphics and pattern recognition, including programming tools and environments.

Weifeng Lyu (Member, IEEE) received the Ph.D. degree in computer science from Beihang University, Beijing, China, in 1998.

He is a Professor, the Dean of the School of Computer Science and Engineering, and the Vice Director of the State Key Laboratory of Software Development Environment, Beihang University. He is also the Leader of the Special Expert Group of Key Technology and Demonstration of Internet of Things and Smart City, Ministry of Science and Technology, Beijing. His research interests include intelligent transportation and data analysis.

Zhang Xiong received the B.S. degree from Harbin Engineering University, Harbin, China, in 1982, and the M.S. degree from Beihang University, Beijing, China, in 1985.

He is a Professor and a Ph.D. Supervisor with the School of Computer Science and Engineering, Beihang University. He is working on computer vision, information security, and data vitalization.

Wei Li (Member, IEEE) received the B.S. degree from the Department of Mathematics and Mechanics, Peking University, Beijing, China, in 1966, and the Ph.D. degree in computer science from the University of Edinburgh, Edinburgh, U.K., in 1983.

He is a Computer Scientist. Since 1966, he has been teaching at Beihang University (formerly, Beijing Institute of Aeronautics), Beijing. He has been a Professor with the School of Computer Science and Engineering, Beihang University since 1986, where he served as a President from 2002 to 2009.

Prof. Li currently serves as the Director of the State Key Laboratory of Software Development Environment. He was elected as a member of the Chinese Academy of Sciences in 1997 and a member of the National Educational Advisory Committee.