Fine-grained Feature Alignment with Part Perspective Transformation for Vehicle ReID

Dechao Meng[1,2], Liang Li[1,*], Shuhui Wang[1], Xingyu Gao[3], Zheng-Jun Zha[4], Qingming Huang[2,1]
[1]Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, CAS, Beijing, China
[2]School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing, China
[3]Institute of Microelectronics, Chinese Academy of Sciences, Beijing, China
[4]School of Information Science and Technology, University of Science and Technology of China, Hefei, China
{dechao.meng,liang.li}@vipl.ict.ac.cn,wangshuhui@ict.ac.cn,gxy9910@gmail.com,zhazj@ustc.edu.cn,qmhuang@ucas.ac.cn

ABSTRACT

Given a query image, vehicle Re-Identification is to search the same vehicle in multi-camera scenarios, which are attracting much attention in recent years. However, vehicle ReID severely suffers from the perspective variation problem. For different vehicles with similar color and type which are taken from different perspectives, all visual patterns are misaligned and warped, which is hard for the model to find out the exact discriminative regions. In this paper, we propose part perspective transformation module (PPT) to map the different parts of vehicle into a unified perspective respectively. The PPT disentangles the vehicle features of different perspectives and then aligns them in a fine-grained level. Further, we propose a dynamically batch hard triplet loss to select the common visible regions of the compared vehicles. Our approach helps the model to generate the perspective invariant features and find out the exact distinguishable regions for vehicle ReID. Extensive experiments on three standard vehicle ReID datasets show the effectiveness of our method.

CCS CONCEPTS

• **Computing methodologies → Visual content-based indexing and retrieval**.

KEYWORDS

vehicle ReID, computer vision, feature alignment, perspective transformation

ACM Reference Format:
Dechao Meng[1,2], Liang Li[1,*], Shuhui Wang[1], Xingyu Gao[3], Zheng-Jun Zha[4], Qingming Huang[2,1]. 2020. Fine-grained Feature Alignment with Part Perspective Transformation for Vehicle ReID. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413573

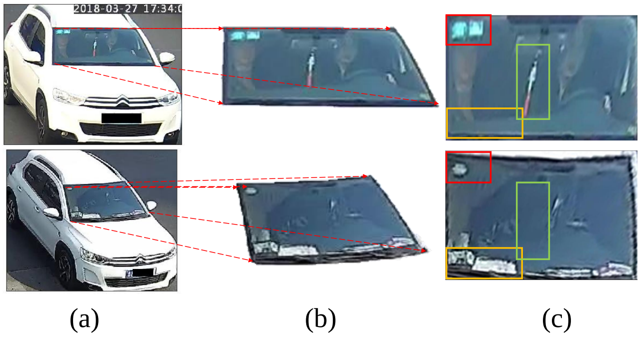

Figure 1: (a). Two different vehicles with the same model. (b). The original cropped window. (c). The window after perspective transformation. The visual cues (marked with boxes) are well aligned in the same position and could be easily compared.

1 INTRODUCTION

Vehicle Re-Identification(vehicle ReID) is to find the same vehicle in cross camera scenarios, which is especially useful when the images are heavily blurred, deformed and occluded. Vehicle ReID has been widely used in the urban surveillance and city security systems [1, 3, 5, 11, 20, 38]. With the proposal of large scale datasets [13, 16, 19] and the development of deep learning [6, 30], researchers have achieved impressive promotion on vehicle ReID.

There are two main challenges in vehicle ReID. The first challenge is that the same vehicle under different perspectives often presents different appearances, which severely reduces the intra-class similarity. To illustrate, vehicles can be regarded as rigid polyhedrons, and under different perspectives, the visible surfaces are different. Directly extracting features from these different surfaces could cause misalignment. The second challenge is the subtle discrepancy of near-duplicate vehicles under the same perspective. Since vehicles are human manufactured products, some of them share the similar models, colors and vehicle types. Even for different vehicles, their visual difference would be subtle when observed from the same perspective.

One potential solution to both problems is to first align the perspective and then compare vehicle images. Following this thought, many works seek to employ alignment in different level to alleviate perspectives variation. Zhou *et al.* [39] proposed an effective multi-view feature inference by an adversarial training architecture. It

[*] Corresponding author.

MM '20, October 12–16, 2020, Seattle, WA, USA
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413573

generated the features in different views and align them in viewpoint level. However, the generated features are inference based on the visible parts, which can not contain the discriminative visual cues.

Recently, researchers introduce local feature extraction to enhance the vehicle features and align them in local region level. Wang *et al.* [31] proposed orientation invariant feature embedding to align the features in keypoints level, where local region features of different orientations were extracted based on 20 key point locations. He *et al.* [5] aligned the features in parts level. It detected the window, lights, and brand for each vehicle through a YOLO detector and align them in feature learning. However, due to perspective transformation, the shape of the same area varies greatly in different perspectives. As seen in the Figure 1, there are two similar vehicles under different perspectives. The distinctive visual cues (annual inspection mark, vehicle decorations) are distributed in the window. Even if we detect and crop the window as Figure 1 (b), the cues are still located in different locations and deformed severely. The previous methods localized different parts and directly extracted features of them, which can not model such variations.

Based on such observations, an intuitive method to handle the feature deformation and misalignment problems is to transform the regions to the same perspective and then compare them to find the discriminative regions. The well-aligned features can alleviate the perspective variation, and help reduce the intra-class difference of same vehicle and increase the discrepancy of near-duplicate vehicles. Perspective transformation is a widely-used linear transform which is to transform a plane from one perspective to another. At the same time, a vehicle can be coarsely regarded as a polyhedron which is composed by different planes. Inspired by this, we propose a part perspective transform (PPT) to map each plane of a vehicle to a unified viewpoint on feature space. First, the vehicles is divided into four different planes based on the predefined vertexes. We solve the perspective transform matrix according to the source vertexes and the target vertexes of vehicle. The source vertexes are generated by the keypoints detection model and the target vertexes are just the vertexes of a rectangle. Then we apply the transformation to the feature maps generated by a deep convolutional network. This transformation restores the region of varied shape to a uniform rectangle in feature space, and well align all the visual cues of vehicle.

However, there is an important problem: vehicle images may be taken from different perspective so that their visible planes are different. In fact, a vehicle can be regarded as a box. In most situations, there are three planes can be seen. For two different images, some planes are visible in both of them. Based on this, we propose a dynamically batch hard triplet loss. The hardest and co-visible planes in a mini-batch are dynamically selected to calculate the loss during the training procedure. The dynamically batch hard triplet loss takes the advantage of the perspective relationship between the images of a mini-batch and aggregate the local features effectively.

The PPT aligns the vehicle features in fine-grained level and the dynamic batch hard triplet loss guides the network to optimize to a right direction. In testing stage, we drop the perspective transform module so it would not introduce any extra computing resources. Our model can be added to any existing vehicle ReID systems.

In summary, our contribution is three fold:

- We proposed a part perspective transformation on feature space to transform the deformed region to a unified perspective, which aligns the features in fine-grained level and guide the model to find the real discriminative regions of vehicles.
- We implement a dynamic batch hard triplet loss strategy to further aggregate and enhance the local features. The loss takes the advantage of the perspective relationship between images in the same batch and guides the network to focus on the co-visible planes of vehicle.
- Extensive experiments on three major benchmarks of vehicle Re-ID show that our method surpasses the SOTA methods on mAP, CMC@1 and CMC@5. Ablation study evaluates the effectiveness of different modules in our model.

2 RELATED WORKS

Vehicle Re-Identification Vehicle ReID is to retrieve the vehicle in a large scale gallery set taken by non-overlapping cameras. It has attracted much attention recently as it serves as an important role in the field of the intelligent transportation systems and smart city [3, 4, 13, 14, 16, 17, 19, 20, 41]. With the development of deep learning, researchers introduce deep learning into vehicle ReID and get impressive results. These methods extract global features by a deep convolutional neural networks, and optimize them by the triplet loss and ID losses. Nonetheless, the features extracted by the neural network directly are unstable due to the variation of illuminate and background. Some methods take advantages of the meta information of the vehicle to overcome these variations, such as vehicle plate [17], attributes [38] and sptio-temporal information [27]. These methods merge the meta information into the features and improves the representation capacity of the features. However, sometimes the meta information are hard to get and if two vehicles share the same color and vehicle type, these methods failed. How to focus on the local distinctive regions remains a problem. He *et al.* [5] detects the vehicle window, brand and lights for local distinctive feature enhancement. This method ignores the visual difference of the vehicle appearance under different viewpoints.

To solve the problem caused by visual difference caused by the perspectives, some methods are proposed[11, 22, 31, 39]. Some researchers begin to focus on the viewpoint variation problem in vehicle ReID recent years. Wang *et al.* [31] has proposed a method generated orientation invariant feature based on pre-defined keypoints detection. In a vehicle image under a definite perspective, some of the keypoints are invisible and they contributed equal to the final features. Khorramshahi *et al.* [11] selects the seven visible keypoints based on a perspective classifier. Zhou *et al.* [39] find a new path to handle the viewpoint problem. It first classify the main perspective of that image and then generate the features of other perspective using a generative scheme. However, the generated features are inferenced based on the known perspective and do not contain the discriminative features. Unlike the above methods, our method first split different planes of the vehicle and then transform each part to a unified perspective, which align the features in a fine level.

Spatial Alignment The idea of conduct perspective transform to align the features and improve the performance on different vision tasks has been widely used [2, 7, 9, 10, 12, 15, 24, 25, 28, 33,

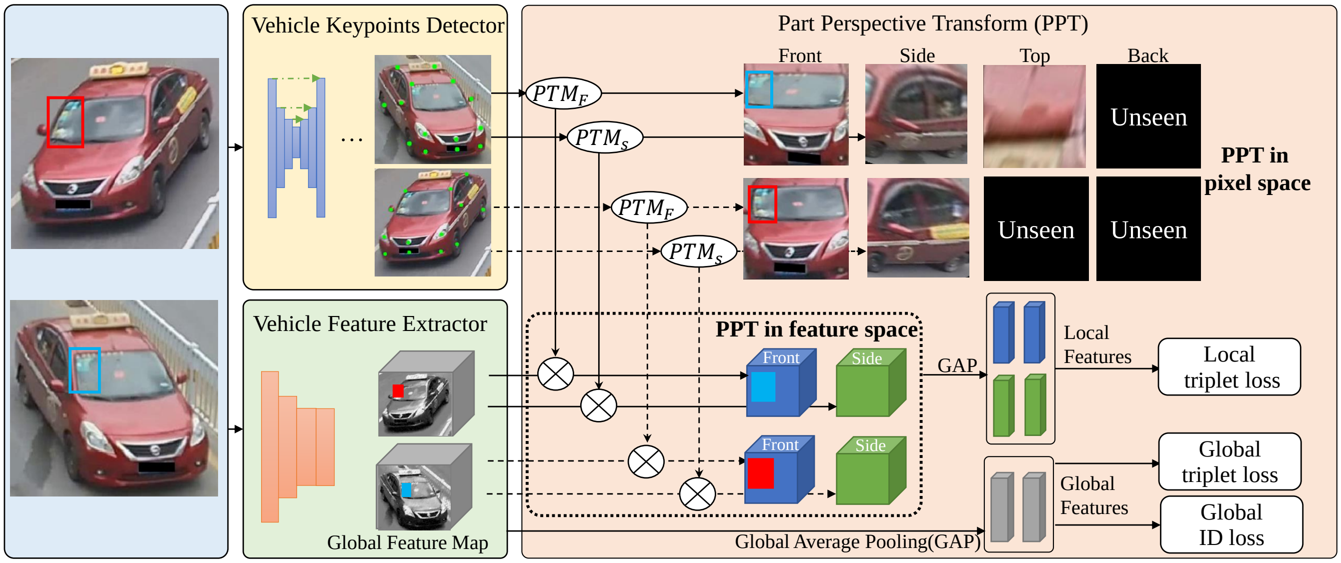

Figure 2: The architecture of our PPT. The red and the blue boxes represents the discriminative regions of a pair of different vehicles. In the original image, they are located in different locations because of the difference in perspective. The part perspective transformation align the each parts of the vehicles to the same perspective. All the features are aligned fine-grained at the same time,hence the discriminative regions are located in the same location. After we get the perspective transform matrix(PTM), we apply it to the feature map to disentangle and align the features of different parts fine-grained. Finnaly, the dynamically triplet loss is applied to the local features to guide the model to focus on the common visible regions.

34, 36, 37]. Spatial Transformer Networks [9, 12] proposed spatial manipulation to learn rotation invariant features in vision recognition tasks. Some researchers [2, 10, 34] use spatial transformer to learn the pose invariant features for the face representation. But the above methods just transform the images to a single perspective. Our part perspective transformation transforms the different parts of the vehicle to different perspectives, which takes the advantages of all the visible parts of a vehicle. Some works [25, 28, 37] align the different parts of the bodies to handle the pose variation problem in person Re-Identification. Compare to person, the vehicles are rigid bodies and there are no pose variations. Our method disentangles the vehicles to different perspectives and aligns them fine-grained.

3 METHODOLOGY

The perspective variation problem is crucial in vehicle ReID, which could cause serious feature mis-alignment and deformation. To address this problem, we propose part perspective transformation to disentangle the different parts of a vehicle and transform them to a unified perspective respectively. We embed the part perspective transform module to a deep neural network. Finally, a dynamically batch hard triplet loss is proposed to handle the problem of missing parts.

3.1 Part Perspective Transformation

Perspective transformation is a linear transform to map a plane from one perspective to another. Let (x, y) denote a point in original image and (x', y') denote corresponding point in the target image. We first augment them to homogeneous coordinates $(x, y, 1)$ and $(x', y', 1)$. The perspective transformation can be formulated as

$$M(x, y, 1)^T = (x', y', 1) \tag{1}$$

where $M \in \mathcal{R}^{3\times3}$ is the transformation matrix. Because of the constrains of the homogeneity, the right-bottom element of M is always equal to 1. Given the coordinates of four pairs of points both in the original image and the target image, we can get the perspective transformation matrix directly by solving the linear equations.

The vehicles can be roughly seen as a polyhedron composed of different planes. As shown in Figure 3, each plane is a trapezoid in their main view. They could be captured in irregular polygons from different perspectives. Given the four vertexes of a polygon, we can transform the polygon to a rectangle by perspective transformation so that the features could be all well aligned.

Instead of mapping the points from source plane to target plane, we reverse the Equation 1 to $M^{-1}(x', y', 1)^T = (x, y, 1)$. For each pixel in the target image, we calculate the corresponding coordinate in the source image. The benefit of the reversal is that for each integer pixel in target image, we can get the corresponding coordinate in original image. As the coordinates of the corresponding points may not be integers, we use bilinear interpolation to get the target features. The bilinear interpolation also makes the back propagation easy to implement.

Now the question is how to find the vertexes in the source image. Wang et al. [31] has annotated 20 vehicle keypoints for VeRi776. The vehicle keypoints are located in the front, left, right, top and back face of the vehicle. Here we just use the annotation of the keypoints. We train a stacked hourglass model [23] to detect the

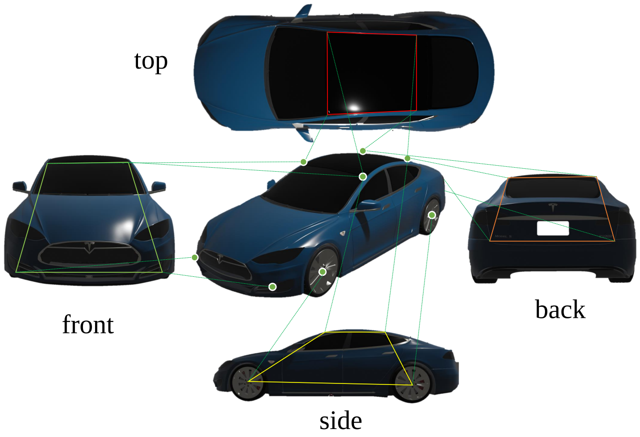

Figure 3: The vehicle keypoints and the transformation regions. The vehicles are split into four different parts based on the different perspectives, which are front, back, side and top. Each part is determined by the four corresponding keypoints.

vehicle keypoints for all datasets. As the vehicles are rigid bodies with the similar shape, the keypoints detection model is not suffering from the deformation problem. Hence the keypoints detection model is generalizable to different vehicle datasets.

We define four planes for a vehicle, namely front, back, side and top, which has been shown in Figure 3. As the left side and right side can not appear in a single image simultaneously and they are usually symmetry, we regard them as one plane and only take the visible one. The planes are defined following the principle that all of the pixels inside the polygon must be approximately in the same plane, which is required by the assumption of Equation 1.

Not all the points are always visible in the image because of the occlusion caused by perspective or the missed keypoints detection. When a point is not visible, all associated parts will be marked as invisible. This will result in a sharp reduction in the number of available planes. In order to solve this problem, we offer some alternative construction rules for the front and back part. For example, if the right-bottom keypoint of the front plane is invisible, we will use the middle-bottom keypoint to take place of it. At the same time, the target point would become the bottom-middle point of the rectangle.

Another failure case is when the points are mis-detected or three of the four keypoints distributed nearly in a line because of specific perspective. In these circumstances, the perspective transformation equation is linear correlation, therefore the perspective transform will fail. To handle this problem, we check whether the polygon generated by the four points is convex and mark the non-convex plane as invisible.

3.2 Network Architecture

We design a deep convolutional neural network with the part perspective transformation (PPT). The overall architecture of the PPT is shown in Figure 2. At the beginning, the images are fed into a feature extractor to get the global feature map. The feature extractor is a deep convolutional neural network without the last fully connected layer. To remain more position information, we set the stride of the last pooling layer to 1. Then the global features are fed into two branches.

The first branch is the global branch. A global average pooling layer is applied to pool the feature map to a single feature vector. After a batch normalization layer, The features are then fed into a classification layer which classifies them to different instances. Following the settings in [21], the features before the BNNeck are used to calculate the distances for the triplet loss and the features after the BNNeck are used for the ID loss.

The other branch is the part perspective transformation branch. First, we train a stacked hourglass network with the keypoints annotation of VeRi776 [31]. Based on the keypoints and the pre-defined four different parts, namely front, back, side and top, we calculate the perspective transformation matrix for each visible part respectively. Second, we apply the part perspective transformation to the global feature map to get local feature maps. The part perspective transformation restores the irregular polygons to rectangles. Third, we use average pooling to each different local feature map to get the final local features. Finally, a triplet loss is followed to them shorten the distance of same instances and enlarge the distance of different instances.

We conduct part perspective transform on the feature space rather than on the raw image. This strategy can bring us three benefits. First, each point of the feature map has a larger receptive field. This will keep more context information for each part. Second, the optimizing process of the local branch can guide the global vehicle feature extractor to focus on the aligned discriminative cues. Hence we can drop the local branch in testing stage. Third, conducting on feature space are more computational effective as we need not to re-calculate the feature map for each part.

The local and global branches share the same convolutional layers. In the training stage, the optimizing procedure of the local branch will guide the backbone to focus on the discriminative regions. In the inference stage, we can drop the local branch and use the global feature only for vehicle retrieval. Under such configuration, the part perspective transformation module can be regarded as a regularizer and would not introduce any extra computation.

3.3 Dynamically Batch Hard Triplet Loss

Batch hard triplet loss [8] has been used in vehicle ReID successfully. It first chooses P different vehicles randomly and then selects K samples for each vehicle to generate a mini-batch. For each vehicle in the mini-batch, it chooses the most similar negative sample and the most dissimilar positive sample in a mini-batch to generate a triplet. Then it optimizes model to shorten the distance between the anchor image and the positive image, and enlarge the distance between the anchor image and negative image. The hardest triplet within a mini-batch is moderate hard compared to the whole dataset, which can avoid the risk of over-fitting on hard samples or wasting time on simple samples.

However, we can not apply it to the local features directly, since different images of a mini-batch may have different visible planes. If that plane is invisible in all other images in that mini-batch, the triplet can not be generated. To handle that problem, we propose dynamically batch hard triplet loss. For a certain vehicle, the invisible planes are not used for triplet loss. For the visible planes, we first determine whether the plane is visible in any positive samples and any negative samples in that mini-batch respectively. If the above conditions are not satisfied, this plane will be dropped. Otherwise we will select the hardest samples from the visible planes.

For an anchor vehicle a, the index of the hardest positive sample in the mini-batch can be formulated as

$$\mathrm{id}_p = \mathrm{argmax}_{\{i|v_i=1,\mathrm{id}_i=\mathrm{id}_a\}} D_{ai} \tag{2}$$

where id_i means the vehicle ID of the ith vehicle in the mini-batch, the v_i denotes the visibility of that part and D_{ai} means the distance of the anchor vehicle and the ith vehicle. In the same way, the index of the hardest negative can be formulated as:

$$\mathrm{id}_n = \mathrm{argmin}_{\{i|v_i=1,\mathrm{id}_i\neq\mathrm{id}_a\}} D_{ai} \tag{3}$$

If there is no visible example in the candidate set, the anchor will be dropped. The triplet loss of that plane is

$$L_i = \{D_{ap} - D_{an} + m\}+ \tag{4}$$

Let L_i^j denote the triplet loss of the ith image, jth plane. The local triplet loss is the sum of anchors and planes.

$$L_l^{\mathrm{triplet}} = \sum_i \sum_j w_j L_i^j \tag{5}$$

w_j is the weights of different planes, which is set to $(0.5, 0.5, 0.3, 0.2)$ for front, back, side and top.

The final optimizing targets of our PPT is composed by the global ID loss, global triplet loss and the local dynamically triplet loss, which can be formulated as

$$L = L_g^{\mathrm{id}} + L_g^{\mathrm{triplet}} + L_l^{\mathrm{triplet}} \tag{6}$$

3.4 Discussion

Differences with PDC Pose-driven deep convolutional model (PDC) [28] use a similar transform strategy for person ReID. It normalize the poses of a person by joint points. But there are still some differences with our method. 1) PDC separated a person to 6 parts based on the joint points and then normalized them to fixed size. The parts were coarsely determined by the joint points and their bounding boxes. In fact, the appearance of some parts change greatly as viewpoint changes, so it is difficult to align features of different parts. Our transformation is based on the vertexes of any irregular quadrilateral. The vehicles are rigid bodies composed by several planes. Each part can be transformed to a unified perspective, so they can be aligned fine-grained. 2) PDC generated a new image using the 6 parts and extracted its features for person ReID. We just transformed the features in the feature space, which needs less computing resources.

Differences with VPM loss Sun et al. [29] proposed a similar loss function in visibility-aware part-level features model (VPM) for person ReID. Both the VPM loss and our proposed loss are to guide the network to focus on the common visible regions of the compared images. There are two main differences. 1) The VPM calculated a soft visibility score for each pre-defined region, which was used to normalize the final distance of all different parts. Then the distance was used to select the hard-triplets directly and to calculate the triplet loss. In our method, we transformed different parts to fixed size and used the hard label to select visible regions. There might be no common visible regions among images of a mini-batch. Our loss dynamically selected the valid parts to generate the hard-triplets for each part respectively. 2) In VPM loss, the smaller area led to a smaller weight in the final distance. But in our loss, all visible regions of different sizes were treated equally. For vehicle images under front-side or back-side view, the area of the side part was usually larger than the front or back parts in a vehicle image, but the discriminative visual cues are often distributed in the back and front parts. Considering this, we ignored the area factor in our loss.

4 EXPERIMENTS

4.1 Datasets

We evaluate our method in three main vehicle ReID benchmarks. In this section, we describe the details of them.

VeRi776 [16] is a the first large scale benchmark for vehicle ReID task. It contains 51032 images of 776 vehicles. The images are taken by 20 cameras with different perspectives, illuminations and occlusions. The training set contains 576 images and the testing set covers the rest 200 vehicles.

VehicleID [13] is a large scale benchmark for vehicle ReID, which contains 113346 images for training and 39647 images for testing. All of the images are under front views and rear views. The testing set is divided into 3 sub testing set with different size (small, medium and large). In the testing stage, One image of each vehicle is selected randomly to form the gallery set and leave the rest images to form the query set. Such testing method results that there are only one ground truth image in the gallery set for each query image. As a result, the cumulated matching characteristics curve is more concerned on this dataset.

VERI-Wild [19] is a recently released dataset and is the largest dataset for vehicle ReID right now. It contains 277797 images for training and 249767 images for testing. The vehicles are taken from 200 cameras across the whole city in a month under all kinds of perspectives, weather and illuminations. Besides, it provides the vehicle brand, color and vehicle type information for researchers to mine the attributes guided methods. Like VehicleID, the testing images are also divided into 3 parts with different size.

4.2 Implementation Details

We implement a stacked hourglass network [23] with 8 stacks and 1 block per stack as the vehicle keypoints detector. We train the network with 6 epochs using a RMSProp optimizer. The learning rate is set to 0.00025.

During the training procedure of vehicle ReID network, the parameters of the vehicle keypoints detection module are fixed. Before fed into the ReID network, the images are first resized to $(256, 256)$, then are padded by 10 pixels in each side. Random cropping is used to crop the image back to $(256, 256)$. After that, random erasing is applied, which has been proved efficiently to prevent over-fitting[21]. We use the ResNet50 [6] pretrained on ImageNet without the classification layer as the feature extractor. An Adam optimizer is used to train the model. We use warmup strategy. The learning rate is set to 3.5e-5 in the first epoch and increased to 3.5e-4

Table 1: Experiments on VeRi776

Method	mAP	CMC@1	CMC@5
OIFE[31]	0.480	0.894	-
VAMI[39]	0.501	-	-
RAM[18]	0.615	0.886	0.940
EALN[20]	0.574	0.844	0.941
AAVER[11]	0.612	0.890	0.947
PRN[5]	0.743	0.943	**0.989**
PPT(ours)	**0.806**	**0.965**	0.983

linearly during the first 10 epochs, then it multiplies 0.1 every 30 epochs. Each mini-batch are composed by 16 different vehicles and each vehicle contains 4 different images. All experiments are conducted on a single GPU with a batch size of 64. For VeRi776 dataset, we train the network for 240 epochs. For the larger VehicleID and VERI-Wild datasets, we train the model for 120 epochs to save the computing resources.

4.3 Comparison with State of the Art

4.3.1 Compared Methods. We compared our methods with some state-of-the-art methods proposed recently for vehicle ReID. Oriented Invariant Feature Embedding network (OIFE) [31] is to detect 20 keypoints and aggregate the local features to enhance the final representation. Viewpoint Aware Multi-view Inference (VAMI) [39] uses viewpoint aware attention to localize the main view and generates the other views of the vehicle in feature space by a generative adversarial network. Region Aware deep Model (RAM) [18] divides the vehicle into different parts evenly and introduces the attributes information into feature learning. Embedding Adversarial Learning Network (EALN) [20] uses generative adversarial network to generate the hard negative samples and cross-view samples to enhance to dataset. Adaptive Attention model for Vehicle ReID (AAVER) [11] detects the keypoints and use viewpoint classification to select the most informative regions. Part Regularized Network (PRN) [5] detects brand, window and lights to guide the network focus on these common discriminative regions.

The selected methods all introduce extra information to enhance the local features for vehicle ReID, like the keypoints information [11, 31], viewpoint information [39], attributes information[18] and bounding box information[5]. All the selected methods use ResNet50 as the backbone.

4.3.2 Results on VeRi776. We evaluate our methods in the VeRi776 dataset and provide the results of three metrics, the mAP, CMC@1 and CMC@5. Table 1 shows the results of our PPT and the compared methods. As can be seen, PPT surpasses most of the counterparts by a large margin in all three metrics. The promising performance is benefited from the fine-grained alignment of local features.

Part Regularized Network (PRN) [5] also pays attention to the local distinguishable regions which is similar to our method. It detects the vehicle brand, window and vehicle lights as the local features to regularizing the ReID network. But the brands, lights and windows are usually not visible when the images are taken in a side perspective. Even when those local regions are detected successfully, the visual appearance would still be misaligned (front light to rear light) and warped because of perspective variation. The part perspective transformation disentangles the features under different perspectives and aligns them fine-grained. Besides, the dynamically triplet loss selects the common visible regions dynamically, which provides an accurate distance. Finally, our method achieves better performance than PRN (6.3% in mAP, 2.2% in CMC@1).

4.3.3 Results on VehicleID. Part perspective transformation can disentangle the features under different perspectives and attend to the common visble regions. In VehicleID, almost all the vehicle images are taken from the front and back view. The features are already aligned for the vehicles under the same view. For the vehicle images under different views (one in front view and one in back view), their common area (such as the roof) is very small. So the promotion on the dataset is not significant. So the potential of our method on this dataset is quiet limited. Nevertheless, our method still achieves better performance in most of the evaluation metrics than the counterparts. This is because even for two images which are both under the front view or both under rear view, feature misalignment still exists because of the subtle difference of the observing angles. Part perspective transform will correct such misalignment, which improves the performance for vehicles under similar viewpoints.

4.3.4 Results on VERI-Wild. Because VERI-Wild is a new proposed dataset, at present, only a few works have reported the performance on that dataset except the original paper [19]. Here, we provide the comparision between our method and the results provided by the original paper for the convenience of later researchers. Here, GoogLeNet[30] is the GoogLeNet model pretrained on CompCar dataset, which is a large scale fine-grained vehicle classification dataset. GSTE[1] proposed group-sensitive-triplet embedding to model the intra-class variance elegantly. FDA-Net [19] used the generative adversarial network to generate the hard negative samples. Our PPT network surpasses the FDA-Net in all evaluation metrics, which shows the ability of fine-grained feature alignment in large scale dataset.

4.4 Ablation Study

4.4.1 The Effectiveness of Different Parts. As is shown in Table 5, we conduct the ablation study of how each part affects the final

Table 2: Experiments on VehicleID

Method	small		medium		large	
	@1	@5	@1	@5	@1	@5
OIFE[31]	-	-	-	-	0.670	0.829
VAMI[39]	0.631	0.833	0.529	0.751	0.473	0.703
RAM[18]	0.752	0.915	0.723	0.870	0.677	0.845
EALN[20]	0.751	0.881	0.718	0.839	0.693	0.814
AAVER[11]	0.747	**0.938**	0.686	0.900	0.635	0.856
PRN[5]	0.784	0.923	0.750	0.883	0.742	0.864
PPTN(ours)	**0.796**	0.923	**0.760**	**0.894**	**0.748**	**0.870**

Table 3: The mAP on VERI-Wild.

Method	small	medium	large
GoogLeNet[32]	0.243	0.242	0.215
Triplet[26]	0.157	0.133	0.099
Softmax[17]	0.264	0.227	0.176
CCL[13]	0.225	0.193	0.148
HDC[35]	0.291	0.248	0.183
GSTE[1]	0.314	0.262	0.195
Unlable-GAN[40]	0.299	0.247	0.182
FDA-Net[19]	0.351	0.298	0.228
PPT	**0.742**	**0.675**	**0.593**

Table 4: The CMC@1 and CMC@5 on VERI-Wild.

Method	small		medium		large	
	@1	@5	@1	@5	@1	@5
GoogLeNet[32]	0.572	0.751	0.532	0.711	0.446	0.636
Triplet[26]	0.447	0.633	0.403	0.590	0.335	0.514
Softmax[17]	0.534	0.750	0.462	0.699	0.379	0.599
CCL[13]	0.570	0.750	0.519	0.710	0.446	0.610
HDC[35]	0.571	0.789	0.496	0.723	0.440	0.649
GSTE[1]	0.605	0.801	0.521	0.749	0.454	0.665
Unlabled Gan[40]	0.581	0.796	0.516	0.744	0.436	0.655
FDA-Net[19]	0.640	0.828	0.578	0.783	0.494	0.705
PPT(ours)	**0.919**	**0.973**	**0.891**	**0.955**	**0.848**	**0.932**

performance on VeRi776 dataset. We gradually increase the number of parts and re-train the model from scratch to see the effectiveness of each parts. The results shows that once we add a kind of region into the training procedure, the overall performance will be improved slightly.

In detail, adding the front part brings 1.3% promotion on mAP. The front part contains the richest distinguishable visual cues, such as annual inspection mark, vehicle decorations. The part perspective transformation can disentangle the front region from the whole vehicle and align the features in fine-grained level, which helps the network find out these visual cues.

Adding the side part can bring 1.2% promotion on mAP. A possible explanation is that in VeRi776, most of the vehicles are taken in front-side view and back-side view. The side view is the common visible view for both perspectives. Traditional methods just ignore the information on the side part and pay more attention to the front and back part. When one of the compared images is under front-side view and the other is the back-side view, the side part would be the only common visible part. The dynamically triplet loss will guide the network to pay more attention to the side part, which provide a more accurate distance measurement.

Table 5: The effectiveness of each parts on VeRi776

front	back	side	roof	mAP	C@1	C@5
				0.774	0.960	0.976
✓				0.787	0.966	0.983
✓	✓			0.788	0.964	0.982
✓	✓	✓		0.800	0.964	0.983
✓	✓	✓	✓	**0.805**	**0.959**	**0.987**

Table 6: Transformation on Different Space

Transformation on	mAP	C@1	C@5
no transformation	0.774	0.960	0.976
pixel space	0.785	0.950	0.982
feature space	**0.805**	**0.959**	**0.987**

4.4.2 Transform on Different Space. Another set of experiments is carried out to verify whether the model can really take the advantages of the perspective transform on feature space. We design an experiment to compare the transformation on feature space and the transformation on pixel space on VeRi776. The experiments of transformation on pixel space is conducted as follows. For each parts, we first conduct the part perspective transformation on the raw images to get four transformed sub images. Then each sub-images are fed into a individual ResNet50 network to get the corresponding local features. Triplet loss and ID loss are both used to optimize the models. Since each model is trained separately, we ensemble the four local models and the global model to get the final results.

Some vehicle parts are invisible in a single vehicle image, so we can not get all of the five features for it. Summing the distance of different models to ensemble the models is impossible. We design an ensemble strategy to handle this problem. First, we assign fixed weights for different models. For the global, front, back, side and top models, we set the weights to $w = [10, 5, 5, 2, 1]$. Second, for two vehicle images to be compared, we first determine the regions that both visible in the two images. Then the corresponding weights are selected and normalized to unit length to keep the scale of the distance. Finally the distance is calculated by the normalized weighted sum of the distance of the common visible parts.

As shown in Table 6, perspective transformation in pixel space is also helpful for the performance because of the well aligned local images provide accurate distances. At the same time, perspective transform in feature space even performs better than the ensemble model. The transformation keeps the information inside the polygons, while there is rich information in the boundary. Transformation in feature space enlarges the receptive field so that the local features can hold more information. Besides, transformation on feature space will also optimize the feature extractor, which brings extra promotion for global branch.

4.5 Qualitative Analysis

4.5.1 How the Part Perspective Feature Works. Classification Activation Map (CAM) is a common tool to observe the activated area of deep neural network for classification. For a specific class, it use the weight connected to the target class in the last fully connected layer as the weights, and generate the heatmap by weighted sum of the convolutional feature map before the global average pooling layer. In ReID task, there is no classification layer in testing stage. We

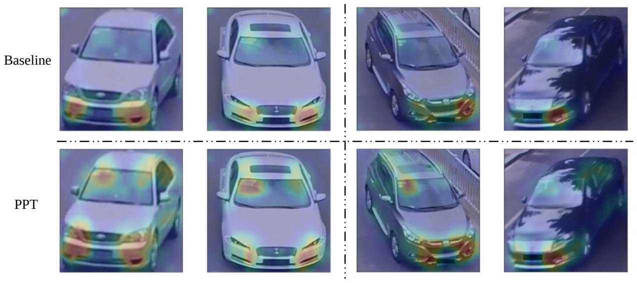

Figure 4: The distance activation map of baseline and our PPT model. The images are divided into four group by the lines. Each group represent a pair of comparison. The top row indicates the heatmap of baseline model and the bottom row indicates the same comparison of our PPT model. Our PPT model attends more on the discriminative regions.

propose Distance Activation Map (DAM) to explore the concerned area of two compared images. For two vehicle features, we use the square of the difference of each dimension as the weights, and then calculate the weighted sum of the feature map of the two images. DAM shows the areas which dominates the distance between the two features. Figure 4 shows two pairs of comparisons.

The images in first row are the DAM of the baseline model and the second row are of our PPT model. First, our PPT model focus more on the discriminative regions such as the annual inspection marks, lights and the windows, while the baseline model only concerns the lights. This is because the fine-grained alignment of the features guide the network find the right different areas. Second, our PPT model pays more attention to the side and roof of the vehicle. This is because the part perspective transform fully exploits the information of different regions, which can prevent the network from over fitting into a single area.

4.5.2 Rank list on different datasets. Figure 5 shows the qualitative results of our method on the three vehicle ReID datasets. We can observe that when the query image and target image are under different views, our PPT can better recognize the same vehicle, which benefits from the disentangle and the fine-grained alignment of features under different perspectives.

The top two rows in Figure 5 shows the results of PPT and PPT without part perspective transform on VeRi776, where PPT ranked all vehicles correctly. The medium two rows are the results on VehicleID. According to the testing rule, for each query, there is only one target image in the gallery set. The key visual cue is the annual mark in the top-left window. PPT find it and rank it in the first place even the perspectives of them are lightly different. The bottom two rows show the results on VERI-Wild. PPT model find the correct vehicle in all kinds of perspectives. We can find that the top-k retrieved vehicles of the baseline model are all of the same perspective and also in similar color and vehicle type. However, PPT find the right images under different perspectives. This indicates that the part perspective transformation can disentangle the features under perspectives and provide the perspective-invariant fine-grained alignment. Dynamically triplet loss guides the network to the common visible regions. They both help the network to find out the exact different regions and find the correct vehicles. Finally, PPT model performs well in different datasets.

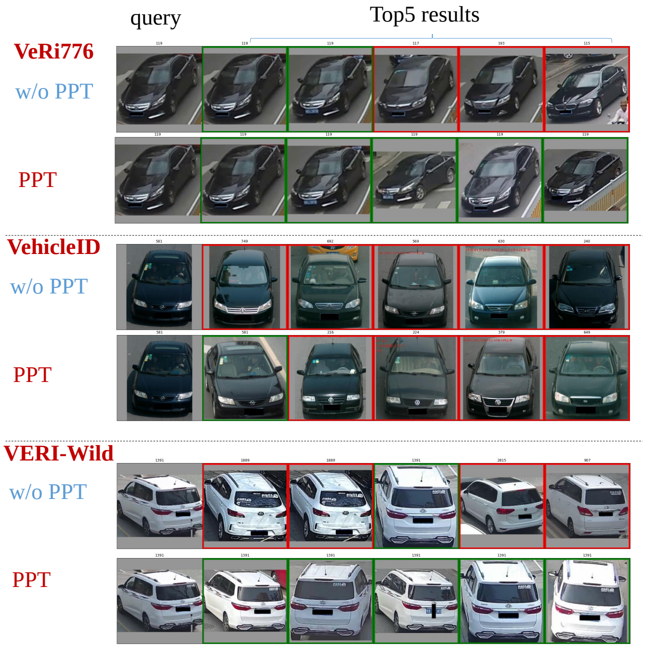

Figure 5: The top five results of the PPT and the baseline model in three datasets.

5 CONCLUSION

In this paper, we introduce the part perspective transformation on feature space to disentangle the features under different perspectives and align them in a fine-grained level for vehicle ReID. The vehicle parts are divided based on the vehicle keypoints, which can disentangle the features of different perspectives. Further the perspective transformation on feature space provides a fine-grained alignment of the vehicle features, which will guide the network to find out the exact discriminative visual cues. We propose the dynamically triplet loss to guide the network to focus on different regions to handle the problem of missing views, which will guide the network to focus on the common visible regions. This not only shortens the distance among intra-instances, but also enlarges the discrepancy of inter-instances. Part perspective transform helps capture the stable and discriminative information of vehicles. The experiments conducted on three datasets show that our model outperforms state-of-the-art methods by a large margin.

6 ACKNOWLEDGEMENT

This work was supported in part by the National Key R&D Program of China under Grant 2018AAA0102003, in part by National Natural Science Foundation of China: 61771457, 61732007, 61620106009, U1636214, 61931008, 61836002, 61672497 and U19B2038, in part by Key Research Program of Frontier Sciences, CAS: QYZDJ-SSW-SYS013.

REFERENCES

[1] Yan Bai, Yihang Lou, Feng Gao, Shiqi Wang, Yuwei Wu, and Ling-Yu Duan. 2018. Group-sensitive triplet embedding for vehicle reidentification. *IEEE Transactions on Multimedia* 20, 9 (2018), 2385–2399.

[2] Dong Chen, Gang Hua, Fang Wen, and Jian Sun. 2016. Supervised transformer network for efficient face detection. In *European Conference on Computer Vision*. Springer, 122–138.

[3] Ruihang Chu, Yifan Sun, Yadong Li, Zheng Liu, Chi Zhang, and Yichen Wei. 2019. Vehicle Re-identification with Viewpoint-aware Metric Learning. In *Proceedings of the IEEE International Conference on Computer Vision*. 8282–8291.

[4] Haiyun Guo, Chaoyang Zhao, Zhiwei Liu, Jinqiao Wang, and Hanqing Lu. 2018. Learning Coarse-to-Fine Structured Feature Embedding for Vehicle Re-Identification. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*. 6853–6860. https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16206

[5] Bing He, Jia Li, Yifan Zhao, and Yonghong Tian. 2019. Part-regularized Near-duplicate Vehicle Re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3997–4005.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. 770–778. https://doi.org/10.1109/CVPR.2016.90

[7] Lingxiao He, Yinggang Wang, Wu Liu, He Zhao, Zhenan Sun, and Jiashi Feng. 2019. Foreground-aware pyramid reconstruction for alignment-free occluded person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*. 8450–8459.

[8] Alexander Hermans, Lucas Beyer, and Bastian Leibe. 2017. In Defense of the Triplet Loss for Person Re-Identification. *CoRR* abs/1703.07737 (2017). arXiv:1703.07737

[9] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. 2015. Spatial transformer networks. In *Advances in neural information processing systems*. 2017–2025.

[10] Amin Jourabloo and Xiaoming Liu. 2017. Pose-invariant face alignment via CNN-based dense 3D model fitting. *International Journal of Computer Vision* 124, 2 (2017), 187–203.

[11] Pirazh Khorramshahi, Amit Kumar, Neehar Peri, Sai Saketh Rambhatla, Jun-Cheng Chen, and Rama Chellappa. 2019. A Dual Path Model With Adaptive Attention For Vehicle Re-Identification. *arXiv preprint arXiv:1905.03397* (2019).

[12] Chen-Hsuan Lin and Simon Lucey. 2017. Inverse compositional spatial transformer networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2568–2576.

[13] Hongye Liu, Yonghong Tian, Yaowei Wang, Lu Pang, and Tiejun Huang. 2016. Deep Relative Distance Learning: Tell the Difference between Similar Vehicles. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. 2167–2175. https://doi.org/10.1109/CVPR.2016.238

[14] Wu Liu, Xinchen Liu, Huadomg Ma, and Peng Cheng. 2017. Beyond human-level license plate super-resolution with progressive vehicle search and domain priori GAN. In *Proceedings of the 25th ACM international conference on Multimedia*. 1618–1626.

[15] Xuejing Liu, Liang Li, Shuhui Wang, Zheng-Jun Zha, Dechao Meng, and Qingming Huang. 2019. Adaptive reconstruction network for weakly supervised referring expression grounding. In *Proceedings of the IEEE International Conference on Computer Vision*. 2611–2620.

[16] Xinchen Liu, Wu Liu, Huadong Ma, and Huiyuan Fu. 2016. Large-scale vehicle re-identification in urban surveillance videos. In *IEEE International Conference on Multimedia and Expo, ICME 2016, Seattle, WA, USA, July 11-15, 2016*. 1–6. https://doi.org/10.1109/ICME.2016.7553002

[17] Xinchen Liu, Wu Liu, Tao Mei, and Huadong Ma. 2018. PROVID: Progressive and Multimodal Vehicle Reidentification for Large-Scale Urban Surveillance. *IEEE Trans. Multimedia* 20, 3 (2018), 645–658. https://doi.org/10.1109/TMM.2017.2751966

[18] Xiaobin Liu, Shiliang Zhang, Qingming Huang, and Wen Gao. 2018. RAM: A Region-Aware Deep Model for Vehicle Re-Identification. In *2018 IEEE International Conference on Multimedia and Expo, ICME 2018, San Diego, CA, USA, July 23-27, 2018*. 1–6. https://doi.org/10.1109/ICME.2018.8486589

[19] Yihang Lou, Yan Bai, Jun Liu, Shiqi Wang, and Lingyu Duan. 2019. Veri-wild: A large dataset and a new method for vehicle re-identification in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3235–3243.

[20] Yihang Lou, Yan Bai, Jun Liu, Shiqi Wang, and Ling - Yu Duan. 2019. Embedding Adversarial Learning for Vehicle Re-Identification. *IEEE Trans. Image Processing* 28, 8 (2019), 3794–3807. https://doi.org/10.1109/TIP.2019.2902112

[21] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. 2019. Bag of tricks and a strong baseline for deep person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 0–0.

[22] Dechao Meng, Liang Li, Xuejing Liu, Yadong Li, Shijie Yang, Zheng-Jun Zha, Xingyu Gao, Shuhui Wang, and Qingming Huang. 2020. Parsing-based View-aware Embedding Network for Vehicle Re-Identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7103–7112.

[23] Alejandro Newell, Kaiyu Yang, and Jia Deng. 2016. Stacked Hourglass Networks for Human Pose Estimation. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII*. 483–499. https://doi.org/10.1007/978-3-319-46484-8_29

[24] Weijian Ruan, Wu Liu, Qian Bao, Jun Chen, Yuhao Cheng, and Tao Mei. 2019. Poinet: pose-guided ovonic insight network for multi-person pose tracking. In *Proceedings of the 27th ACM International Conference on Multimedia*. 284–292.

[25] M Saquib Sarfraz, Arne Schumann, Andreas Eberle, and Rainer Stiefelhagen. 2018. A pose-sensitive embedding for person re-identification with expanded cross neighborhood re-ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 420–429.

[26] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 815–823.

[27] Yantao Shen, Tong Xiao, Hongsheng Li, Shuai Yi, and Xiaogang Wang. 2017. Learning Deep Neural Networks for Vehicle Re-ID with Visual-spatio-Temporal Path Proposals. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. 1918–1927. https://doi.org/10.1109/ICCV.2017.210

[28] Chi Su, Jianing Li, Shiliang Zhang, Junliang Xing, Wen Gao, and Qi Tian. 2017. Pose-driven deep convolutional model for person re-identification. In *Proceedings of the IEEE international conference on computer vision*. 3960–3969.

[29] Yifan Sun, Qin Xu, Yali Li, Chi Zhang, Yikang Li, Shengjin Wang, and Jian Sun. 2019. Perceive Where to Focus: Learning Visibility-aware Part-level Features for Partial Person Re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 393–402.

[30] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. 1–9. https://doi.org/10.1109/CVPR.2015.7299023

[31] Zhongdao Wang, Luming Tang, Xihui Liu, Zhuliang Yao, Shuai Yi, Jing Shao, Junjie Yan, Shengjin Wang, Hongsheng Li, and Xiaogang Wang. 2017. Orientation Invariant Feature Embedding and Spatial Temporal Regularization for Vehicle Re-identification. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. 379–387. https://doi.org/10.1109/ICCV.2017.49

[32] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. 2015. A large-scale car dataset for fine-grained categorization and verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3973–3981.

[33] Shijie Yang, Liang Li, Shuhui Wang, Weigang Zhang, Qingming Huang, and Qi Tian. 2019. Skeletonnet: A hybrid network with a skeleton-embedding process for multi-view image representation learning. *IEEE Transactions on Multimedia* 21, 11 (2019), 2916–2929.

[34] Junho Yim, Heechul Jung, ByungIn Yoo, Changkyu Choi, Dusik Park, and Junmo Kim. 2015. Rotating your face using multi-task deep neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 676–684.

[35] Yuhui Yuan, Kuiyuan Yang, and Chao Zhang. 2017. Hard-Aware Deeply Cascaded Embedding. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. 814–823. https://doi.org/10.1109/ICCV.2017.94

[36] Beichen Zhang, Liang Li, Shijie Yang, Shuhui Wang, Zheng-Jun Zha, and Qingming Huang. 2020. State-Relabeling Adversarial Active Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8756–8765.

[37] Liming Zhao, Xi Li, Yueting Zhuang, and Jingdong Wang. 2017. Deeply-learned part-aligned representations for person re-identification. In *Proceedings of the IEEE international conference on computer vision*. 3219–3228.

[38] Aihua Zheng, Xianmin Lin, Chenglong Li, Ran He, and Jin Tang. 2019. Attributes Guided Feature Learning for Vehicle Re-identification. *arXiv preprint arXiv:1905.08997* (2019).

[39] Yi Zhou and Ling Shao. 2018. Viewpoint-Aware Attentive Multi-View Inference for Vehicle Re-Identification. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. 6489–6498. https://doi.org/10.1109/CVPR.2018.00679

[40] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.

[41] Yangchun Zhu, Zheng-Jun Zha, Tianzhu Zhang, Jiawei Liu, and Jiebo Luo. 2020. A Structured Graph Attention Network for Vehicle Re-Identification.. In *Proceedings of the 28th ACM international conference on Multimedia*.