VehicleNet: Learning Robust Visual Representation for Vehicle Re-Identification

Zhedong Zheng, Tao Ruan, Yunchao Wei, Yi Yang, and Tao Mei

***Abstract*—One fundamental challenge of vehicle re-identification (re-id) is to learn robust and discriminative visual representation, given the significant intra-class vehicle variations across different camera views. As the existing vehicle datasets are limited in terms of training images and viewpoints, we propose to build a unique large-scale vehicle dataset (called VehicleNet) by harnessing four public vehicle datasets, and design a simple yet effective two-stage progressive approach to learning more robust visual representation from VehicleNet. The first stage of our approach is to learn the generic representation for all domains (i.e., source vehicle datasets) by training with the conventional classification loss. This stage relaxes the full alignment between the training and testing domains, as it is agnostic to the target vehicle domain. The second stage is to fine-tune the trained model purely based on the target vehicle set, by minimizing the distribution discrepancy between our VehicleNet and any target domain. We discuss our proposed multi-source dataset VehicleNet and evaluate the effectiveness of the two-stage progressive representation learning through extensive experiments. We achieve the state-of-art accuracy of 86.07% mAP on the private test set of AICity Challenge, and competitive results on two other public vehicle re-id datasets, i.e., VeRi-776 and VehicleID. We hope this new VehicleNet dataset and the learned robust representations can pave the way for vehicle re-id in the real-world environments.**

***Index Terms*—Vehicle re-identification, image representation, convolutional neural networks.**

I. Introduction

VEHICLE re-identification (re-id) is to spot the car of interest in different cameras and is usually viewed as a sub-task of image retrieval problem [1]. It could be applied to the public place for the traffic analysis, which facilitates the traffic jam management and the flow optimization [2]. Yet vehicle re-id remains challenging since it inherently contains multiple intra-class variants, such as viewpoints, illumination and occlusion. Thus, vehicle re-id system demands a robust and discriminative visual representation given that the realistic scenarios are diverse and complicated. Recent years, Convolutional Neural Network (CNN) has achieved the state-of-the-art performance in many computer vision tasks, including person re-id [3]–[5] and vehicle re-id [6]–[8], but CNN is data-hungry and prone to over-fitting small-scale datasets. Since the paucity of vehicle training images compromises the learning of robust features, vehicle re-id for the small datasets turn into a challenging problem.

One straightforward approach is to annotate more data and re-train the CNN-based model on the augmented dataset. However, it is usually unaffordable due to the annotation difficulty and the time cost. Considering that many vehicle datasets collected in lab environments are publicly available, an interesting problem arises: Can we leverage the public vehicle image datasets to learn the robust vehicle representation? Given vehicle datasets are related and vehicles share the similar structure, more data from different sources could help the model to learn the common knowledge of vehicles. Inspired by the success of large-scale datasets, e.g., ImageNet [9], we collect a large-scale vehicle dataset, called VehicleNet.

Intuitively, we could utilize VehicleNet to learn the relevance between different vehicle re-id datasets. Then the robust features could be obtained by minimizing the objective function. However, different datasets are collected in different environments, and contains different biases. Some datasets, such as CompCar [10], are mostly collected in the car exhibitions, while other datasets, e.g., City-Flow [2] and VeRi-776 [6], are collected in the real traffic scenarios. Thus, another scientific problem of how to leverage the multi-source vehicle dataset occurs. In several existing works, some researchers resort to transfer learning [11], which aims at transferring the useful knowledge from the labeled source domain to the unlabeled target domain and minimizing the discrepancy between the source domain and the target domain. Inspired by the spirit of transfer learning, in this work, we propose a simple two-stage progressive learning strategy to learn from VehicleNet and adapt the trained model to the realistic environment.

In a summary, to address the above-mentioned challenges, i.e., the data limitation and the usage of multi-source dataset, we propose to build a large-scale dataset, called VehicleNet, via the public datasets and learn the common knowledge of the vehicle representation via two-stage progressive learning (see Fig. 1). Specifically, instead of only using the original training dataset, we first collect free vehicle images from the web.

Manuscript received April 7, 2020; revised June 16, 2020; accepted July 31, 2020. Date of publication August 7, 2020; date of current version August 24, 2021. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Junwei Han. *(Corresponding author: Yi Yang.)*

Zhedong Zheng, Yunchao Wei, and Yi Yang are with the Australian Artificial Intelligence Institute, University of Technology Sydney, NSW 2007, Australia (e-mail: zhedong.zheng@student.uts.edu.au; yunchao.wei@uts.edu.au; yi.yang@uts.edu.au).

Tao Ruan is with the Institute of Information Science at Beijing Jiaotong University, and the Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing 100044, China (e-mail: 16112064@bjtu.edu.cn).

Tao Mei is with the AI Research of JD.COM, Beijing 100105, China (e-mail: tmei@live.com).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2020.3014488

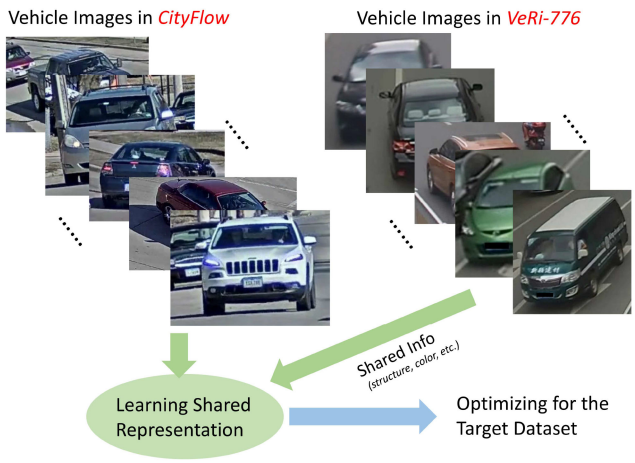


Fig. 1. The motivation of our vehicle re-identification method by leveraging public datasets. The common knowledge of discriminating different vehicles could be transferred to the final model.

Comparing with the training set of the CityFlow dataset, we scale up the number of training images from 26,803 to 434,440 as a new dataset called VehicleNet. We train the CNN-based model to identify different vehicles, and extract features. With the proposed two-stage progressive learning, the model is further fine-tuned to adapt to the target data distribution, yielding the performance boost. In the experiment, we show that it is feasible to train models with a combination of multiple datasets. When training the model with more samples, we observe a consistent performance boost, which is consistent with the observation in some recent works [1], [12], [13]. Without explicit vehicle part matching or attribute recognition, the CNN-based model learns the viewpoint-invariant feature by "seeing" more vehicles. Albeit simple, the proposed method achieves mAP 75.60% on the private testing set of CityFlow [2] without extra information. With the temporal and spatial annotation, our method further arrives the 86.07% mAP. The result surpasses the AICity Challenge champion, who also uses the temporal and spatial annotation. In a nutshell, our contributions are two-folds:

- To address the data limitation, we introduce one large-scale dataset, called VehicleNet, to borrow the strength of the public vehicle datasets, which facilitate the learning of robust vehicle features. In the experiment, we verify the feasibility and effectiveness of learning from VehicleNet.
- To leverage the multi-source vehicle images in VehicleNet, we propose a simple yet effective learning strategy, i.e., the two-stage progressive learning approach. We discuss and analyze the effectiveness of the two-stage progressive learning approach. The proposed method has achieved competitive performance on the CityFlow benchmark as well as two public vehicle re-identification datasets, i.e., VeRi-776 [6] and VehicleID [14].

II. Related Work

A. Vehicle Re-identification

Vehicle re-identification (re-id) demands robust and discriminative image representation. The recent progress of vehicle re-identification has been due to two aspects: 1) the availability of the new vehicle datasets [2], [6], [14], [15] and 2) the discriminative vehicle feature from deeply-learned models.

Zapletal *et al.* [16] first collect a large-scale dataset with vehicle pairs and extract the color histograms and oriented gradient histograms feature to discriminate different cars. With recent advance in Convolutional Neural Network (CNN), Liu *et al.* [17] combine the CNN-based feature with the traditional hand-crafted features to obtain the robust feature. Qian *et al.* [18] and Guo *et al.* [19] propose to aggregate the multi-level feature to enrich the representation. To take fully advantages of the fine-grained patterns, Wang *et al.* [8] first explore the vehicle structure and then extract the part-based CNN features according to the location of key points. Besides, Shen *et al.* [20] involve the temporal-spatial information into the model training as well as the inference process. Another line of works regards vehicle re-identification as a metric learning problem, and explore the objective functions to help the representation learning. Triplet loss has been widely studied in person re-id [21]–[23], and also has achieved successes in the vehicle re-id [6]. Zhang *et al.* [24] further company the classification loss with triplet loss, which further improves the re-identification ability. Furthermore, Yan *et al.* [15] propose a multi-grain ranking loss to discriminate the appearance-similar cars. Besides, some works also show the attributes, e.g., color, manufactories and wheel patterns, could help the model to learn the discriminative feature [2], [25], [26].

B. Dataset Augmentation

Many existing works focus on involving more samples to boost the training. One line of works leverage the generative model to synthesize more samples for training. Wu *et al.* [27] and Yue *et al.* [28] propose to transfer the image into different image styles, e.g., weather conditions, and learn the robust feature for semantic segmentation. In a similar spirit, Zheng *et al.* [1], [29] utilize the Generative Adversarial Network (GAN) [30] to obtain lots of pedestrian images, and then involve the generated samples into training as an extra regularization term. Another line of works collects the real-world data from Internet to augment the original dataset. One of the pioneering work [12] is to collect large number of images via searching the keywords on the online engine, i.e., Google. After removing the noisy data, the augmented dataset facilitate the model to achieve the state-of-the-art performance on several fine-grained datasets, e.g., CUBird [31]. In a similar spirit, Zheng *et al.* [32] exploit noisy photos of university buildings from Google, benefiting the model learning. In contrast with these existing works, we focus on leveraging the public datasets with different data biases to learn the common knowledge given that vehicles share the similar structure.

C. Transfer Learning

Transfer learning is to propagate the knowledge of the source domain to the target domain [11]. On one hand, several recent works focus on the alignment between the source domain and the target domain, which intend to minimize the discrepancy of two domains. One of the pioneering works [33] is to apply

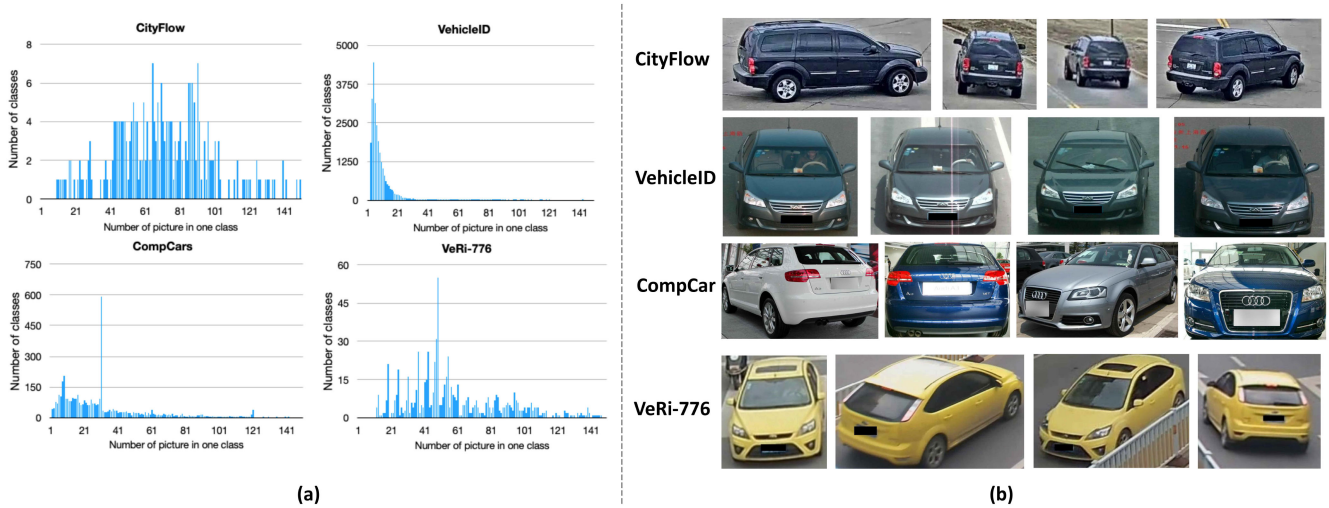


Fig. 2. (a) The image distribution per class in the vehicle re-id datasets, e.g., CityFlow [2], VehicleID [14], CompCar [10], and VeRi-776 [6]. We observe that the two largest datasets, i.e., VehicleID and CompCars, suffer from the limited images per class. (b) Here we also provide the image samples of the four datasets. The four datasets contain different visual biases, such as illumination conditions, and collection places and viewpoints.

the cyclegan [34] to transfer the image style to the target domain, and then train the model on the transferred data. In this way, the model could learn the similar patterns of the target data. Besides the pixel-level alignment, some works [35]–[37] focus on aligning the network activation in the middle or high layers of the neural network. The discriminator is deployed to discriminate the learned feature of source domain from that of target domain, and the main target is to minimize the feature discrepancy via adversarial learning. On the other hand, some works deploy the pseudo label learning, yielding competitive results as well [38], [39]. The main idea is to make the model more confident to the prediction, which minimizes the information entropy. The pseudo label learning usually contains two steps. The first step is to train one model from scratch on the source domain and generate the pseudo label for the unlabeled data. The second step is to fine-tune the model and make the model adapt to the target data distribution via the pseudo label. Inspired by the existing works, we propose one simple yet effective two-stage progressive learning. We first train the model on the large-scale VehicleNet dataset and then finetune the model on the target dataset. The proposed method is also close to the traditional pre-training strategy, but the proposed method could converge quickly and yield competitive performance due to the related vehicle knowledge distilled in the model.

III. Dataset Collection and Task Definition

A. Dataset Analysis

We involve four public datasets, i.e., CityFlow [2], VeRi-776 [6], CompCar [10] and VehicleID [14] into training. It results in 434,440 training images of 31,805 classes as **VehicleNet**. Note that four public datasets are collected in different places. There are no overlapping images with the validation set or the private test set. We plot the data distribution of all four datasets in Fig. 2. **CityFlow** [2] is one of the largest vehicle re-id datasets. There are bounding boxes of 666 vehicle identities annotated. All images are collected from 40 cameras in a realistic scenario. We follow the official training/test protocol, which results in 36,935 training images of 333 classes and 19,342 testing images of other 333 classes. The training set is collected from 36 cameras, and test is collected from 23 cameras. There are 19 overlapping cameras. Official protocol does not provide a validation set. We therefore further split the training set into a validation set and a small training set. After the split, the training set contains 26,803 images of 255 classes, and the validation query set includes 463 images of the rest 78 classes. We deploy the original training set as the gallery of the validation set. **VeRi-776** [6] contains 49,357 images of 776 vehicles from 20 cameras. The dataset is collected in the real traffic scenario, which is close to the setting of CityFlow. The author also provides the meta data, e.g., the collected time and the location. **CompCar** [10] is designed for the fine-grained car recognition. It contains 136,726 images of 1,716 car models. The author provides the vehicle bounding boxes. By cropping and ignoring the invalid bounding boxes, we finally obtain 136,713 images for training. The same car model made in different years may contain the color and shape difference. We, therefore, view the same car model produced in the different years as different classes, which results in 4,701 classes. **VehicleID** [14] consists 2211,567 images of 26,328 vehicles. The vehicle images are collected in two views, i.e., frontal and rear views. Despite the limited viewpoints, the experiment shows that VehicleID also helps the viewpoint-invariant feature learning. **Other Datasets** We also review other public datasets of vehicle images in Table I. Some datasets contain limited images or views, while others lack ID annotations. Therefore, we do not use these datasets, which may potentially compromise the feature learning.

B. Task Definition

Vehicle re-identification aims to learn a projection function F, which maps the input image x to the discriminative representation $f_i = F(x_i)$. Usually, F is decided by minimizing the following optimization function on a set of training data

TABLE I
PUBLICLY AVAILABLE VEHICLE DATASETS. †: WE VIEW THE VEHICLE MODEL PRODUCED IN DIFFERENT YEARS AS DIFFERENT CLASSES, WHICH LEADS TO MORE CLASSES. ‡: THE DOWNLOADED IMAGE NUMBER IS SLIGHTLY DIFFERENT WITH THE REPORT NUMBER IN [14]

Datasets	# Cameras	# Images	#IDs
CityFlow [2]	40	56,277	666
VeRi-776 [6]	20	49,357	776
CompCar [10] †	n/a	136,713	4,701
VehicleID [14] ‡	2	221,567	26,328
PKU-VD1 [15]	1	1,097,649	1,232
PKU-VD2 [15]	1	807,260	1,112
VehicleReID [40]	2	47,123	n/a
PKU-Vehicle [41]	n/a	10,000,000	n/a
StanfordCars [42]	n/a	16,185	196
VehicleNet	**62**	434,440	**31,805**

$X = \{x_i\}_{i=1}^N$ with the annotated label $Y = \{y_i\}_{i=1}^N$:

$$\min \sum_{i=1}^{N} loss(WF(x_i), y_i) + \alpha\Omega(F), \tag{1}$$

where $loss(\cdot,\cdot)$ is the loss function, W is the weight of the classifier, $\Omega(F)$ is the regularization term, and α is the weight of the regularization.

Our goal is to leverage the augmented dataset for learning robust image representation given that the vehicle shares the common structure. The challenge is to build the vehicle representation which could fit the different data distribution among multiple datasets. Given $X^d = \{x_i^d\}_{i=1}^N$ with the annotated label $Y^d = \{y_i^d\}_{i=1,d=1}^N$, the objective could be formulated as:

$$\min \sum_{d=1}^{D} \sum_{i=1}^{N} loss(WF(x_i^d), y_i^d) + \alpha\Omega(F), \tag{2}$$

where D is the number of the augmented datasets. The loss demands F could be applied to not only the target dataset but also other datasets, yielding the good scalability. In terms of the regularization term $\Omega(F)$, we adopt the common practise of weight decay as weight regularization, which prevents the weight value from growing too large and over-fits the dataset.

IV. METHODOLOGY

A. Model Structure

Feature Extractor: Following the common practise in re-identification problems [6], [43], we deploy the off-the-shelf Convolutional Neural Network (CNN) model pre-trained on the ImageNet dataset [44] as the backbone. Specifically, the proposed method is scalable and could be applied to different network backbones. We have trained and evaluated the state-of-the-art structures, including ResNet-50 [45], DenseNet-121 [46], SE-ResNeXt101 [47] and SENet-154 [47], in the Section V. The classification layer of the pre-trained backbone model is removed, which is dedicated for image recognition on ImageNet. The original average pooling layer is replaced with the adaptive average pooling layer, and the adaptive average pooling layer outputs the mean of the input feature map in terms of the height and width channels. We add one fully-connected layer '*fc1*' of 512 dimensions and one batch normalization layer to reduce the feature dimension, followed by a fully-connected layer '*fc2*' to output the final classification prediction as shown in the Fig. 3. The length of the classification prediction equals to the category number of the dataset. The cross-entropy loss is to penalize the wrong vehicle category prediction.

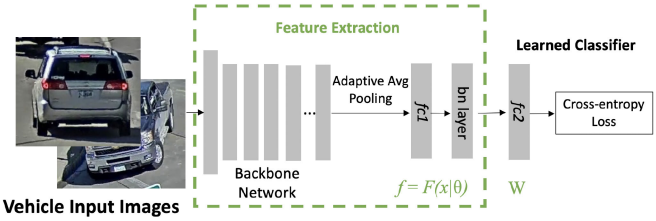


Fig. 3. Illustration of the model structure. We remove the original classifier of the ImageNet pre-trained model, add a new classifier and replace the average pooling with the adaptive average pooling layer. The adaptive average pooling is to squeeze the output to the pre-defined shape (i.e., 1×1).

Feature Embedding: Vehicle re-identification is to spot the vehicle of interest from different cameras, which demands a robust representation to various visual variants, e.g., viewpoints, illumination and resolution. Given the input image x, we intend to obtain the feature embedding $f = F(x|\theta)$. In this work, the CNN-based model contains the projection function F and one linear classifier. Specifically, we regard the '*fc2*' as the conventional linear classifier with the learnable weight W, and the module before the final classifier as F with the learned parameter θ. The output of the batch normalization layer as f (see the green box in the Fig. 3). When inference, we extract the feature embedding of query images and gallery images. The ranking list is generated according to the similarity with the query image. Given the query image, we deploy the cosine similarity, which could be formulated as $s(x_n, x_m) = \frac{f_n}{||f_n||_2} \times \frac{f_m}{||f_m||_2}$. The $||.||_2$ denotes l^2 norm of the corresponding feature embedding. The large similarity value indicates that the two images are highly relevant.

B. Two-Stage Progressive Learning

The proposed training strategy contains two stages. During the first stage, we train the CNN-based model on the VehicleNet dataset and learn the general representation of the vehicle images. In particular, we deploy the widely-adopted cross-entropy loss in the recognition tasks, and the model learns to identify the input vehicle images from different classes. The loss could be formulated as:

$$L_{ce} = \sum_{i=1}^{N} -p_i \log(q_i), \tag{3}$$

where p_i is the one-hot vector of the ground-truth label y_i. The one-hot vector $p_i(c) = 1$ if the index c equals to y_i, else $p_i(c) = 0$. q_i is the predicted category probability of the model, and $q_i = WF(x_i|\theta)$. Since we introduce the multi-source dataset, the cross-entropy loss could be modified to work with the multi-source data.

$$L_{ce} = \sum_{d=1}^{D} \sum_{i=1}^{N} -p_i^d \log(q_i^d), \tag{4}$$

where d denotes the index of the public datasets in the proposed VehicleNet. Specifically, $d = 1, 2, 3, 4$ denotes the four datasets

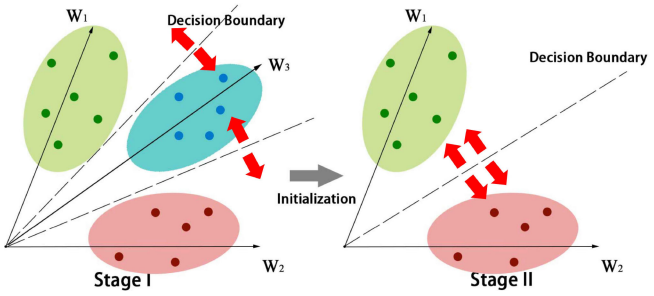


Fig. 4. Geometric Interpretation. Here we give a three-class sample to show our intuition. W_i denotes the class weight of the final linear classifier. In this example, the third class denotes one auxiliary class, which belongs to VehicleNet but the target domain. Therefore, in the Stage-II fine-tuning, we remove the auxiliary classes, including W_3. The cross-entropy loss of Stage-I pulls the samples with the same label together (close to either the relative weight W_1, W_2 or W_3). In this way, the positive pair is closer than the negative pair, while the samples are far from the decision boundary. Stage I, therefore, leads to a decent weight initialization to be used in Stage II with a large margin from decision boundary, when we leave out the auxiliary class, i.e., the third class with W_3, from VehicleNet.

in VehicleNet, i.e., CityFlow [2], VehicleID [14], CompCar [10] and VeRi-776 [6], respectively. p_i^d is the one-hot vector of y_i^d, and $q_i^d = WF(x_i^d|\theta)$. Note that we treat all the dataset equally, and demand the model with good scalability to data of different datasets in VehicleNet.

In the first stage, we optimize the Equation (4) on all the training data of VehicleNet to learn the shared representation for vehicle images. The Stage-I model is agnostic to the target environment, hence the training domain and the target domain are not fully aligned. In the second stage, we take one more step to further fine-tune the model only upon the target dataset, e.g., CityFlow [2], according to the Equation 3. In this way, the model is further optimized for the target environment. Since only one dataset is considered in the Stage-II and the number of vehicle category is decreased, in particular, the classifier is replaced with the new $fc2$ layer with 333 classes from CityFlow. To preserve the learned knowledge, only the classification layer of the trained model is replaced. Although the new classifier is learned from scratch, attribute to the decent initial weights in the first stage, the model could converge quickly and meets the demand for quick domain adaptation. We, therefore, could stop the training at the early epoch. To summarize, we provide the training procedure of the proposed method in Algorithm 1.

Discussion: What are the advantages of the proposed two-stage progressive learning? First, the learned representation is more robust. In the Stage-I, we demand the model could output the discriminative representation for all of the data in the multi-source VehicleNet. The model is forced to learn the shared knowledge among the training vehicle images, which is similar to the pre-training practise in many re-ID works [5], [21]. Second, the representation is also more discriminative. The first stage contains 31,805 training classes during training. The auxiliary classes of other real vehicles could be viewed as "virtual class" as discussed in [48]. Here we provide one geometric interpretation in the Fig. 4. After the convergence of Stage I, the cross-entropy loss pulls the data with the same label together, and pushes the data from different labels away from each other on the either side of the decision boundary. In this manner, as shown in the Fig. 4 (right), the first stage will provide better weight initialization for the subsequent fine-tuning on the target dataset. It is because the auxiliary classes expand the decision space and the data is much far from the new decision boundary, yielding discriminative features.

Algorithm 1: Training Procedure of the Proposed Method

Require: The multi-source VehicleNet dataset $X^d = \{x_i^d\}_{i=1}^D$; The corresponding label $Y^d = \{y_i^d\}_{i=1}^D$;

Require: The initialized model parameter θ; The first stage iteration number T_1 and the second stage iteration number T_2.

1: **for** $iteration = 1$ to T_1 **do**

2: Stage-I: Input x_t^j to $F(\cdot|\theta)$, extract the prediction of the classifier, and calculate the cross-entropy loss according to Equation 4:

$$L_{ce} = \sum_{d=1}^{D}\sum_{i=1}^{N} -p_i^d \log(q_i^d), \tag{5}$$

where p_i^d is the one-hot vector of y_i^d, and q_i^d is the predict probability. $q_i^d = WF(x_i^d|\theta)$, W is the final fully-connected layer, which could be viewed as a linear classifer. We update the θ and W during the training.

3: **end for**

4: **for** $iteration = 1$ to T_2 **do**

5: Stage-II: We further fine-tune the trained model only on the target dataset, e.g., CityFlow. The classifier is replaced with a new one, since we have less classes. We assume that CityFlow is the first dataset ($d = 1$). Thus, we could update θ upon the cross-entropy loss according to Equation 3:

$$L_{ce} = \sum_{i=1}^{N} -p_i^1 \log(q_i^1). \tag{6}$$

6: where p_i^1 is the one-hot vector of y_i^1 of the CityFlow dataset, and q_i^1 is the predict probability. $q_i^1 = W'F(x_i^1|\theta)$. We note that W' is the new fully-connected layer, which is trained from scratch and different from W used in the Stage-I.

7: **end for**

8: **return** θ.

C. Post-Processing

Furthermore, we apply several post-processing techniques during the inference stage as shown in Fig. 5.

Cropped Images: We notice that the vehicle datasets usually provide a relatively loose bounding box, which may introduce the background noise. Therefore, we re-detect vehicles with the state-of-the-art MaskRCNN [49]. For the final result, the vehicle representation is averaged between original images and cropped images, yielding more robust vehicle representations.

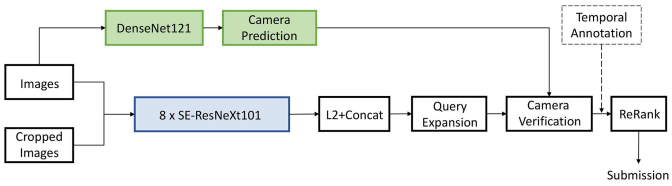


Fig. 5. The inference pipeline for AICity Challenge Competition. Given one input image and the corresponding cropped image via MaskRCNN [49], we extract features from the trained models, i.e., 8×SE-ResNeXt101 [47]. We normalize and concatenate the features. Meanwhile, we extract the camera prediction from the camera-aware model, i.e., the fine-tuned DenseNet121 [46]. Then query expansion and camera verification are applied. Finally, we utilize the re-ranking technique [50] to retrieve more positive samples.

Model Ensemble: We adopt a straightforward late-fusion strategy, i.e., concatenating features [5]. Given the input image x_i, the embedding f_i^j denotes the extracted feature of x_i from the j-th trained model. The final pedestrian descriptor could be represented as: $f_i = [\frac{f_i^1}{||f_i^1||_2}, \frac{f_i^2}{||f_i^2||_2}, \ldots \frac{f_i^n}{||f_i^n||_2}]$. The $||\cdot||_2$ operator denotes l^2-norm, and $[\cdot]$ denotes feature concatenation.

Query Expansion & Re-ranking: We adopt the unsupervised clustering method, i.e., DBSCAN [51] to find the most similar samples. The query feature is updated to the mean feature of the other queries in the same cluster. Furthermore, we adopt the re-ranking method [50] to refine the final result, which takes the high-confidence candidate images into consideration. In this work, our method does not modify the re-ranking procedure. Instead, the proposed method obtains discriminative vehicle features that distill the knowledge from "seeing" various cars. With better features, re-ranking is more effective.

Camera Verification: We utilize the camera verification to further remove some hard-negative samples. When training, we train one extra CNN model, i.e., DenseNet121 [46], to recognize the camera from which the photo is taken. When testing, we extract the camera-aware features from the trained model and then cluster these features by DBSCAN [51]. In this way, we could obtain clustering centers. We applied the prior assumption that the query image and the true matches are taken in different cameras, indicating that the query images and true matches in the gallery usually belong to different camera clustering centers. Given a query image, we remove the images of the same camera cluster from candidate images.

Temporal Annotation: Temporal annotation can be easily obtained by recording the timestamp of which the target vehicle passes by. The prior assumption is that the vehicles usually appear once in the whole camera network, indicating that the two images with long time interval belong to two different vehicles. Given the timestamp t of the query image, we filter out the image in the gallery with long interval τ. As a result, we only consider the candidate images with the timestamp in $[t-\tau, t+\tau]$, which also could filter out lots of the hard-negative samples.

V. Experiment

A. Implementation Details

For two widely-adopted public datasets, i.e., VeRi-776 and VehicleID, we follow the setting in [61], [62] to conduct a fair comparison. We adopt ResNet-50 [63] as the backbone network and input images are resized to 256×256. We apply SGD optimizer with momentum of 0.9 and mini-batch size of 36. The weight decay is set to 0.0001 following the setting in [63]. The initial learning rate is set to 0.02 and is divided by a factor 10 at the 40-th epoch of the first stage and the 8-th epoch in the second stage. The total epochs of the first stage is 60 epochs, while the second-stage fine-tuning is trained with 12 epochs. **When inference, we only apply the mean feature of the image flipped horizontally, without using other post-processing approaches for two academic datasets.**

For the competition dataset, i.e., CityFlow [2], we adopt one sophisticated model, i.e., SE-ResNeXt101 [47] as the backbone to conduct the ablation study and report the performance. The vehicle images are resized to 384×384. Similarly, the first stage is trained with 60 epochs, and the second stage contains 12 epochs. When conducting inference on the validation set, we only apply the mean feature of the image flipped horizontally, without using other post-processing approaches. In contrast, to achieve the best results on the private test set of CityFlow, we apply all the post-processing methods mentioned in Section IV-C.

TABLE II
The Rank@1 (%) and mAP (%) Accuracy with Different Number of Training Images. Here We Report the Results Based on the Validation Set we Splitted. † Note That We Split a Validation Set From the Training Set, Which Leads to Less Training Data

Training Datasets	# Training Images	Performance	
		Rank@1 (%)	mAP (%)
CityFlow [2] †	26,803	73.65	37.65
CityFlow [2]+ VeRi-776 [6]	+49,357	79.48	43.47
CityFlow [2]+ CompCar [10]	+136,713	83.37	48.71
CityFlow [2]+ VehicleID [14]	+221,567	83.37	47.56
VehicleNet	434,440	**88.77**	**57.35**

B. Qualitative Results

Effect of VehicleNet: To verify the effectiveness of the public vehicle data towards the model performance, we involve different vehicle datasets into training and report the results, respectively (see Table II). There are two primary points as follows: First, the model performance has been improved by involving the training data of one certain datasets, either VeRi-776, CompCar or VehicleID. For instance, the model trained on CityFlow + CompCar has achieved 83.37% Rank@1 and 48.71% mAP, which surpasses the baseline of 73.65% Rank@1 and 37.65% mAP. It shows that more training data from other public datasets indeed helps the model learning the robust representation of vehicle images. Second, we utilize the proposed large-scale VehicleNet to train the model, which contains all the training data of four public datasets. We notice that there are +15.12% Rank@1 improvement from 73.65% Rank@1 to 88.77% Rank@1, and +19.70% mAP increment from 37.65% mAP to 57.35% mAP. It shows that the proposed VehicleNet has successfully "borrowed" the strength from multiple datasets and help the model learning robust and discriminative features.

Comparison with the State-of-the-art: We mainly compare the performance with other methods on the test sets of two public

TABLE III
COMPARISON WITH THE STATE-OF-THE-ART METHODS IN TERMS OF RANK@1 (%) AND MAP (%) ACCURACY ON THE VERI-776 DATASET [6] AND THE VEHICLEID DATASET [14]. -: DENOTES THE CONVENTIONAL HAND-CRAFTED FEATURES AND *: DENOTES THAT THE APPROACH UTILIZES THE SELF-DESIGNED NETWORK STRUCTURE. THE BEST RESULTS ARE IN **BOLD**

Methods	Backbones	VeRi-776		VehicleID (Small)		VehicleID (Medium)		VehicleID (Large)	
		mAP (%)	Rank@1 (%)	Rank@1 (%)	Rank@5 (%)	Rank@1 (%)	Rank@5 (%)	Rank@1 (%)	Rank@5 (%)
LOMO [52]	-	9.78	23.87	19.74	32.14	18.95	29.46	15.26	25.63
GoogLeNet [10]	GoogLeNet	17.81	52.12	47.90	67.43	43.45	63.53	38.24	59.51
FACT [6]	-	18.73	51.85	49.53	67.96	44.63	64.19	39.91	60.49
XVGAN [53]	*	24.65	60.20	52.89	80.84	-	-	-	-
SiameseVisual [20]	*	29.48	41.12	-	-	-	-	-	-
OIFE [8]	*	48.00	65.92	-	-	-	-	67.0	82.9
VAMI [7]	*	50.13	77.03	63.12	83.25	52.87	75.12	47.34	70.29
NuFACT [54]	*	53.42	81.56	48.90	69.51	43.64	65.34	38.63	60.72
FDA-Net [55]	*	55.49	84.27	-	-	59.84	77.09	55.53	74.65
QD-DLF [56]	*	61.83	88.50	72.32	92.48	70.66	88.90	68.41	83.37
AAVER [57]	ResNet-50	58.52	88.68	72.47	93.22	66.85	89.39	60.23	84.85
PVSS [58]	ResNet-50	62.62	90.58	-	-	-	-	-	-
C2FRank [19]	GoogLeNet	-	-	61.1	63.5	56.2	60.0	51.4	53.0
VANet [59]	GoogLeNet	66.34	89.78	83.26	95.97	81.11	**94.71**	77.21	**92.92**
PAMTRI [60]	DenseNet-121	71.88	92.86	-	-	-	-	-	-
SAN [61]	ResNet-50	72.5	93.3	79.7	94.3	78.4	91.3	75.6	88.3
Part [62]	ResNet-50	74.3	94.3	78.4	92.3	75.0	88.3	74.2	86.4
Ours (Stage-I)	ResNet-50	80.91	95.95	83.26	96.77	81.13	93.68	79.06	91.84
Ours (Stage-II)	ResNet-50	**83.41**	**96.78**	**83.64**	**96.86**	**81.35**	93.61	**79.46**	92.04

vehicle re-id datasets, i.e., VeRi-776 [6] and VehicleID [14] as well as AICity Challenge [60] private test set. The comparison results with other competitive methods are as follows: **VeRi-776 & VehicleID:** There are two lines of competitive methods. One line of works deploy the hand-crafted features [6], [52] or utilize the self-designed network [7], [8], [54]. In contrast, another line of works leverages the model pre-trained on ImageNet, yielding the superior performance [57], [59], [60], [62]. As shown in Table III, we first evaluate the proposed approach on the VeRi-776 dataset [6]. We leave out the VeRi-776 test set from the VehicleNet to fairly compare the performance, and we deploy the ResNet-50 [63] as backbone network, which is used by most compared methods. The proposed method has achieved 83.41% mAP and 96.78% Rank@1 accuracy, which is superior to the second best method, i.e., Part-based model [62] (74.3% mAP and 94.3% Rank@1) by a large margin. Meanwhile, we observe a similar result on the VehicleID dataset [14] in all three settings (Small /Medium /Large). Small, Medium and Large setting denotes different gallery sizes of 800, 1600 and 2400, respectively. The proposed method also arrives competitive results, e.g., 83.64% Rank@1 of the small gallery setting, 81.35% Rank@1 of the medium gallery setting, and 79.46% Rank@1 of the large gallery setting. One competitive method, VANet [59], has achieved comparable results on VehicleID, but is inferior to the proposed method on VeRi-776. It is because VANet introduces one extra viewpoint module, which could discriminate different viewpoints, i.e., front view and rear view. Since the VehicleID dataset only contains two views, VANet works well. In contrast, on another benchmark VeRi-776, containing 20 cameras, the proposed method is more scalable than VANet in terms of the multi-camera scenario. **AICity Challenge.** For AICity Challenge Competition (on the private test set of CityFlow [2]), we adopt a slightly different training strategy, using the large input size as well as the model ensemble. The images are resized to 384×384. We adopt the mini-batch SGD with the weight decay of 5e-4 and a momentum of 0.9. In the first stage, we decay the learning rate of 0.1 at the 40-th and 55-th epoch. We trained 32 models with different batchsizes and different learning rates. In the second stage, we fine-tune the models on the original dataset. We decay the learning rate of 0.1 at the 8-th epoch and stop training at the 12-th epoch. Finally, we select 8 best models on the validation set to extract the feature. When testing, we adopt the horizontal flipping and scale jittering, which resizes the image with the scale factors $[1, 0.9, 0.8]$ to extract features. As a result, we arrive at 75.60% mAP on the private testing set. Without extra temporal annotations, our method has already achieved competitive results (see Table IV). With the help of extra annotation of temporal and spatial information, we have achieved 86.07% mAP, which surpasses the champion of the AICity Vehicle Re-id Challenge 2019.

TABLE IV
COMPETITION RESULTS OF AICITY VEHICLE RE-ID CHALLENGE ON THE PRIVATE TEST SET. OUR RESULTS ARE IN **BOLD**

Team Name	Temporal Annotation	mAP(%)
Baidu_ZeroOne [64]	✓	85.54
UWIPL [65]	✓	79.17
ANU [66]	✓	75.89
Ours	×	**75.60**
Ours	✓	**86.07**

C. Further Evaluations and Discussion

Effect of Two-stage Progressive Learning: We compare the final results of the Stage I and the Stage II on the private test set of CityFlow (see Table V) . We do not evaluate the performance on the validation set we splitted, since we utilize all training images into fine-tuning. The model of Stage II has arrived 87.45% Rank@1 and 75.60% mAP accuracy, which has significantly surpassed the one of Stage I +7.39% mAP and +4.75% Rank@1. It verifies the effectiveness of the two-stage learning. In the Stage I, the target training set, i.e., CityFlow, only occupy 6% of VehicleNet. The learned model, therefore, is

TABLE V
THE RANK@1(%) AND MAP (%) ACCURACY WITH DIFFERENT STAGES ON THE CITYFLOW PRIVATE TEST SET

	Private Test Set	
	Rank@1(%)	mAP(%)
Stage I	82.70	68.21
Stage II	87.45	75.60

TABLE VI
EFFECT OF DIFFERENT POST-PROCESSING TECHNIQUES ON THE CITYFLOW VALIDATION SET

Method	Performance					
with Cropped Image?		✓	✓	✓	✓	✓
Model Ensemble?			✓	✓	✓	✓
Query Expansion?				✓	✓	✓
Camera Verification?					✓	✓
Re-ranking?						✓
mAP (%)	57.35	57.68	61.29	63.97	65.97	74.52

TABLE VII
THE RANK@1 (%) AND MAP (%) ACCURACY WITH DIFFERENT BACKBONES ON THE CITYFLOW VALIDATION SET. THE BEST RESULTS ARE IN **BOLD**

Backbones	ImageNet	Performance	
	Top5(%)	Rank@1 (%)	mAP (%)
ResNet-50 [63]	92.98	77.97	43.65
DenseNet-121 [46]	92.14	83.15	47.17
SE-ResNeXt101 [47]	95.04	**83.37**	**48.71**
SENet-154 [47]	95.53	81.43	45.14

TABLE VIII
THE RANK@1(%) AND MAP (%) ACCURACY ON THE CITYFLOW VALIDATION SET WITH TWO DIFFERENT SAMPLING METHODS. HERE WE USE THE RESNET-50 BACKBONE

Sampling Policy	Performance	
	Rank@1(%)	mAP(%)
Naive Sampling	77.97	43.65
Balanced Sampling	76.03	40.09

sub-optimal for the target environment. To further optimize the model for CityFlow, the second stage fine-tuning helps to minor the gap between VehicleNet and the target set, yielding better performance. We also observe similar results on the other two datasets, i.e., VeRi-776 and VehicleID. As shown in the last two row of Table III, the Stage-II fine-tuning could further boost the performance. For instance, the proposed method has achieved $+2.50\%$ mAP and $+0.83\%$ Rank@1 improvement on VeRi-776. We compare the two-stage learning strategy with the domain adaption policy, which is usually based on style transferring. Specifically, we apply the prevailing CycleGAN [34] to change the style of data in VehicleNet to VeRi-776. We observe that CycleGAN could successfully change the vehicle style. However, CycleGAN introduces some unrealistic noise. As shown in Table IX, the style transferring method is inferior to the proposed two-stage learning strategy. We speculate that it is due to the generation noise by CycleGAN. Besides, training CycleGAN costs extra time, which may be not ideal for the fast domain adaptation.

Effect of Part-based Method Fusion: The proposed method has the potential to fuse with other competitive methods. We

TABLE IX
COMPARISON WITH OTHER COMPLEMENTARY METHODS ON VERI-776

Method	Rank@1(%)	mAP(%)
w CycleGAN data	92.91	75.23
Stage I	95.95	80.91
Stage II	96.78	83.41
Stage II + PCB [3]	97.26	83.54

select the second best method [61] on VeRi-776 to verify the potential of the proposed method. [61] utilizes one similar policy as PCB [3] to split the feature map horizontally into 4 parts. As shown in Table IX, ours + PCB can take one step further, yielding 97.26% Rank@1 and 83.54% mAP.

Effect of Post-processing: Here we provide the ablation study of post-processing techniques on the validation set of CityFlow (see Table VI). When applying the augmentation with cropped images, model ensemble, query expansion, camera verification and re-ranking, the performance gradually increases, which verifies the effectiveness of post-processing methods.

Effect of Different Backbones: We observe that different backbones may lead to different results. As shown in Table VII, SE-ResNeXt101 [47] arrives the best performance with 83.37 Rank@1 and 48.71% mAP on the validation set of the CityFlow dataset. We speculate that it is tricky to optimize some large-scale neural networks due to the problem of gradient vanishing. For instance, we do not achieve a better result (45.14% mAP) with SENet-154 [47], which preforms better than SE-ResNeXt101 [47] on ImageNet [9]. We hope this observation could help the further study of the model backbone selection in terms of the re-identification task.

Effect of Sampling Policy: Since we introduce more training data in the first stage, the data sampling policy has a large impact on the final result. We compare two sampling policies. The naive method is to sample every image once in every epoch. Another method is called balanced sampling policy. The balanced sampling is to sample the images of different class with equal possibility. As shown in Table VIII, the balanced sampling harms the result. We speculate that the long-tailed data distribution (see Fig. 2) makes the balanced sampling have more chance to select the same image in the classes with fewer images. Thus the model is prone to over-fit the class with limited samples, which compromise the final performance. Therefore, we adopt the naive sampling policy.

Visualization of Vehicle Re-id Results: As shown in Fig. 6, we provide the qualitative image search results on CityFlow. We select the four query images from different viewpoints, i.e., the front view, the overhead view, the rear view and the side view. The proposed method has successfully retrieved the relevant results in the top-5 of the ranking list.

Visualization of Learned Heatmap: Following [41], [67], we utilize the network activation before the pooling layer to visualize the attention of the learned model. For instance, given one middle-level feature of $14 \times 14 \times 2048$, we aggregate the activation of all channels via summation, resulting one feature of 14×14. Then we normalize the feature to [0, 1], and map the value to the corresponding heatmap color. The generation code

Fig. 6. Qualitative image search results using the vehicle query images from the CityFlow dataset. We select the four query images from different viewpoints. The results are sorted from left to right according to the similarity score. The true-matches are in green, when the false-matches are in red.

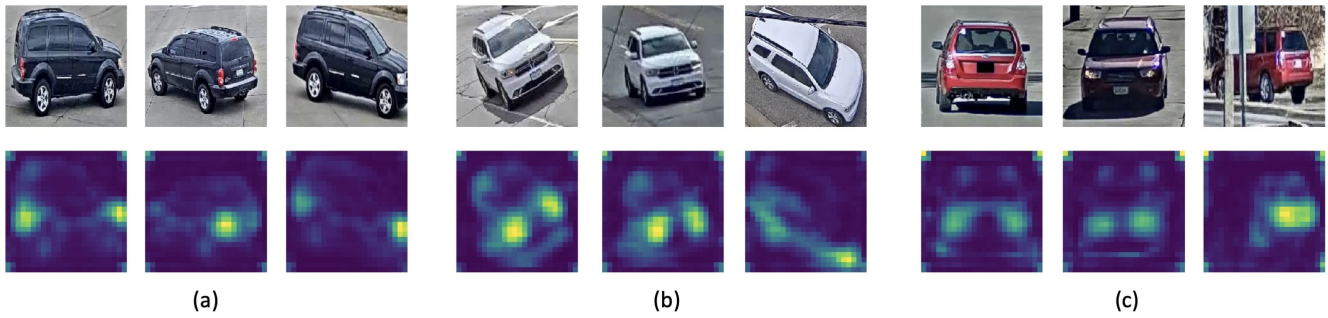


Fig. 7. Visualization of the activation heatmap in the learned model on VehicleNet. The vehicle images in every subfigure (a)-(c) are from the same vehicle ID. Noted that there do exist strong response values at the regions containing discriminative details, such as headlights and tire types.

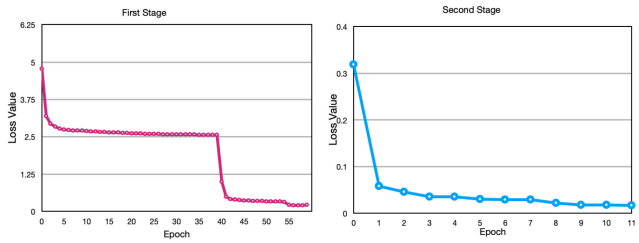


Fig. 8. The training losses of the two stages. Due to the large-scale data and classes, the first stage (left) takes more epochs to converge. Attribute to the trained weight of the first stage, the second stage (right) converge early.

is avaiable at.[1] As shown in Fig. 7, the trained model has strong response values at the regions containing discriminative details, such as headlights and tire types. In particular, despite different viewpoints, the model could focus on the salient areas, yielding the viewpoint-invariant feature.

Model Convergence: As shown in Fig. 8 (left), despite a large number of training classes, i.e., 31,805 categories in VehicleNet, the model could converge within 60 epochs. As discussed, the first stage provides a decent weight initialization for fine-tuning in the second stage. Therefore, Stage-II training converges quickly within 12 epochs (see Fig. 8 (right)).

Time Cost: The Stage-I training costs about 30 hours on the whole VehicleNet with 3 × Nvidia 2080TI. The Stage-II training costs about 1.5 hours for fine-tuning.

[1]https://github.com/layumi/Person_reID_baseline_pytorch/blob/dev/visual_heatmap.py

VI. Conclusion

In this paper, we intend to address two challenges in the context of vehicle re-identification, i.e., the lack of training data, and how to harness multiple public datasets. To address the data limitation, we build a large-scale dataset called VehicleNet. To learn the robust feature, we propose a simple yet effective approach, called two-stage progressive learning. We achieve 86.07% mAP accuracy in AICity19 Challenge and competitive performance on two other public datasets, i.e., VeRi-776 and VehicleID.

In the future, we will try two data collection methods to further improve the work. 1). One method is to collect data from the search engine, i.e., Google, to enlarge the dataset. The existing works [12], [32] show that a few noise annotations usually do not compromise the model training. 2). The other way is to generate the synthetic data by either GAN [30] or 3D-models [68], to further explore the robust representation learning. Besides, we will explore weakly supervised learning approaches [69]–[71] to fully take advantage of unlabeled data.

References

[1] Z. Zheng *et al.*, "Joint discriminative and generative learning for person re-identification," *Comput. Vis. Pattern Recognit.*, 2019.
[2] T. Zheng *et al.*, "Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification," in *Proc. Comput. Vis. Pattern Recognit.*, 2019.
[3] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, "Beyond part models: Person retrieval with refined part pooling," in *Proc. Europ. Conf. Comput. Vis.*, 2018.
[4] Y. Sun, L. Zheng, W. Deng, and S. Wang, "SVDNet for pedestrian retrieval," in *Proc. Int. Conf. Comput. Vis.*, 2017.

[5] Z. Zheng, L. Zheng, and Y. Yang, "Pedestrian alignment network for large-scale person re-identification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 10, Oct. 2019.

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

[7] Y. Zhou and L. Shao, "Aware attentive multi-view inference for vehicle re-identification," in *Proc. Comput. Vis. Pattern Recognit.*, 2018.

[8] Z. Wang *et al.*, "Orientation invariant feature embedding and spatial temporal regularization for vehicle re-identification," in *Proc. Int. Conf. Comput. Vis.*, 2017.

[9] J. Deng *et al.*, "Imagenet: A large-scale hierarchical image database," in *Proc. Comput. Vis. Pattern Recognit.*, 2009.

[10] L. Yang, P. Luo, C. Change Loy, and X. Tang, "A large-scale car dataset for fine-grained categorization and verification," in *Proc. Comput. Vis. Pattern Recognit.*, 2015.

[11] S. J. Pan and Q. Yang, "A survey on transfer learning," *TKDE*, vol. 22, no. 10, pp. 1345–1359, 2009.

[12] J. Krause *et al.*, "The unreasonable effectiveness of noisy data for fine-grained recognition," in *Europ. Conf. Comput. Vis.*, 2016.

[13] D. Mahajan *et al.*, "Exploring the limits of weakly supervised pretraining," in *Proc. Europ. Conf. Comput. Vis.*, 2018.

[14] H. Liu, Y. Tian, Y. Wang, L. Pang, and T. Huang, "Deep relative distance learning: Tell the difference between similar vehicles," in *Proc. Comput. Vis. Pattern Recognit.*, 2016.

[15] K. Yan, Y. Tian, Y. Wang, W. Zeng, and T. Huang, "Exploiting multi-grain ranking constraints for precisely searching visually-similar vehicles," in *Int. Conf. Comput. Vis.*, 2017.

[16] D. Zapletal and A. Herout, "Vehicle re-identification for automatic video traffic surveillance," in *Proc. Comput. Vis. Pattern Recognit. Workshops*, 2016.

[17] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2016.

[18] X. Qian, Y. Fu, Y.-G. Jiang, T. Xiang, and X. Xue, "Multi-scale deep learning architectures for person re-identification," *Proc. Comput. Vis. Pattern Recognit.*, 2017.

[19] H. Guo, C. Zhao, Z. Liu, J. Wang, and H. Lu, "Learning coarse-to-fine structured feature embedding for vehicle re-identification," in *Proc. AAAI*, 2018.

[20] 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. Int. Conf. Comput. Vis.*, 2017.

[21] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," 2017, *arXiv:1703.07737*.

[22] Z. Zheng *et al.*, "Dual-path convolutional image-text embeddings with instance loss," *ACM Trans. Multimedia Comput. Commun., Appl.*, vol. 16, no. 2, pp. 1–23, 2020.

[23] Y. Ding, H. Fan, M. Xu, and Y. Yang, "Adaptive exploration for unsupervised person re-identification," *ACM Trans. Multimedia Comput., Commun. Appl.*, vol. 16, no. 1, pp. 1–19, 2020.

[24] Y. Zhang, D. Liu, and Z.-J. Zha, "Improving triplet-wise training of convolutional neural network for vehicle re-identification," in *Proc. Integr. Computat. Mater. Eng.*, 2017.

[25] Y. Lin *et al.*, "Improving person re-identification by attribute and identity learning," *Pattern Recognit.*, vol. 95, pp. 151–161, 2019.

[26] J. Wang, X. Zhu, S. Gong, and W. Li, "Transferable joint attribute-identity deep learning for unsupervised person re-identification," in *Proc. Comput. Vis. Pattern Recognit.*, 2018.

[27] Z. Wu, X. Wang, J. E. Gonzalez, T. Goldstein, and L. S. Davis, "Ace: Adapting to changing environments for semantic segmentation," in *Proc. Int. Conf. Comput. Vis.*, 2019.

[28] X. Yue *et al.*, "Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data," in *Proc. Int. Conf. Comput. Vis.*, 2019.

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

[30] I. Goodfellow *et al.*, "Generative adversarial nets," in *NeurIPS*, 2014.

[31] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200-2011 dataset," California Institute of Technology, Tech. Rep. CNS-TR-2011-001, 2011.

[32] Z. Zheng, Y. Wei, and Y. Yang, "University-1652: A multi-view multi-source benchmark for drone-based geo-localization," *ACM Multimedia*, 2020.

[33] J. Hoffman *et al.*, "Cycada: Cycle-consistent adversarial domain adaptation," *ICML*, 2018.

[34] J.-Y. Zhu *et al.*, "Toward multimodal image-to-image translation," in *Proc. NeurIPS*, 2017.

[35] Y.-H. Tsai *et al.*, "Learning to adapt structured output space for semantic segmentation," in *Comput. Vis. Pattern Recognit.*, 2018.

[36] Y.-H. Tsai, K. Sohn, S. Schulter, and M. Chandraker, "Domain adaptation for structured output via discriminative patch representations," in *Proc. Int. Conf. Comput. Vis.*, 2019.

[37] Y. Luo, L. Zheng, T. Guan, J. Yu, and Y. Yang, "Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation," in *Proc. Comput. Vis. Pattern Recognit.*, 2019.

[38] Y. Zou, Z. Yu, V. Kumar, and J. Wang, "Unsupervised domain adaptation for semantic segmentation via class-balanced self-training," in *Proc. Europ. Conf. Comput. Vis.*, 2018.

[39] H.-Y. Lee, H.-Y. Tseng, J.-B. Huang, M. Singh, and M.-H. Yang, "Diverse image-to-image translation via disentangled representations," in *Proc. Europ. Conf. Comput. Vis.*, 2018.

[40] D. Zapletal and A. Herout, "Vehicle re-identification for automatic video traffic surveillance," in *Proc. Comput. Vis. Pattern Recognit.*, 2016.

[41] Y. Bai *et al.*, "Group-sensitive triplet embedding for vehicle reidentification," *TMM*, vol. 20, no. 9, pp. 2385–2399, 2018.

[42] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3d object representations for fine-grained categorization," in *3DRR*, 2013.

[43] Z. Zhedong *et al.*, "Joint discriminative and generative learning for person re-identification," in *Proc. Comput. Vis. Pattern Recognit.*, 2019.

[44] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[45] L. He, J. Liang, H. Li, and Z. Sun, "Deep spatial feature reconstruction for partial person re-identification: Alignment-free approach," in *Proc. Comput. Vis. Pattern Recognit.*, 2018.

[46] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. Comput. Vis. Pattern Recognit.*, 2017.

[47] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. Comput. Vis. Pattern Recognit.*, 2018.

[48] B. Chen, W. Deng, and H. Shen, "Virtual class enhanced discriminative embedding learning," in *Proc. NeurIPS*, 2018.

[49] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *Proc. Int. Conf. Comput. Vis.*, 2017.

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

[51] M. Ester *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Knowl. Discovery Databases*, 1996.

[52] S. Liao, Y. Hu, X. Zhu, and S. Li, "Person re-identification by local maximal occurrence representation and metric learning," in *Proc. Comput. Vis. Pattern Recognit.*, 2015.

[53] 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.

[54] X. Liu, W. Liu, T. Mei, and H. Ma, "Provid: Progressive and multimodal vehicle reidentification for large-scale urban surveillance," *IEEE Trans. Multimedia*, vol. 20, no. 3, pp. 645–658, Mar. 2018.

[55] Y. Lou, Y. Bai, J. Liu, S. Wang, and L. Duan, "Veri-wild: A large dataset and a new method for vehicle re-identification in the wild," in *Proc. Comput. Vis. Pattern Recognit.*, 2019.

[56] J. Zhu *et al.*, "Vehicle re-identification using quadruple directional deep learning features," *IEEE Trans. Intell. Transp. Syst.*, 2019.

[57] P. Khorramshahi *et al.*, "A dual path modelwith adaptive attention for vehicle re-identification," 2019, *arXiv:1905.03397*.

[58] X.-C. Liu, H.-D. Ma, and S.-Q. Li, "Pvss: A progressive vehicle search system for video surveillance networks," *J. Comput. Sci. Technol.*, vol. 34, no. 3, pp. 634–644, 2019.

[59] R. Chu *et al.*, "Vehicle re-identification with viewpoint-aware metric learning," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 8282–8291.

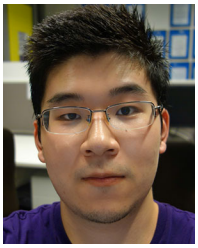
[60] Z. Tang *et al.*, "Pamtri: Pose-aware multi-task learning for vehicle re-identification using highly randomized synthetic data," in *Proc. Int. Conf. Comput. Vis.*, 2019.

[61] J. Qian, W. Jiang, H. Luo, and H. Yu, "Stripe-based and attribute-aware network: A two-branch deep model for vehicle re-identification," 2019, *arXiv:1910.05549*.

[62] B. He, J. Li, Y. Zhao, and Y. Tian, "Part-regularized near-duplicate vehicle re-identification," in *Proc. Comput. Vis. Pattern Recognit.*, 2019.

[63] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. Comput. Vis. Pattern Recognit.*, 2016.

[64] X. Tan *et al.*, "Multi-camera vehicle tracking and re-identification based on visual and spatial-temporal features," in *Comput. Vis. Pattern Recognit. Workshops*, 2019.
[65] T.-W. Huang, J. Cai, H. Yang, H.-M. Hsu, and J.-N. Hwang, "Multi-view vehicle re-identification using temporal attention model and metadata re-ranking," in *Proc. Comput. Vis. Pattern Recognit. Workshops*, 2019.
[66] K. Lv *et al.*, "Vehicle reidentification with the location and time stamp," in *Proc. Comput. Vis. Pattern Recognit. Workshops*, 2019.
[67] Z. Zheng, L. Zheng, and Y. Yang, "A discriminatively learned cnn embedding for person reidentification," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 14, no. 1, pp. 1–20, 2017.
[68] Y. Yao, L. Zheng, X. Yang, M. Naphade, and T. Gedeon, "Simulating content consistent vehicle datasets with attribute descent," 2019, *arXiv:1912.08855*.
[69] D. Zhang, J. Han, L. Zhao, and D. Meng, "Leveraging prior-knowledge for weakly supervised object detection under a collaborative self-paced curriculum learning framework," *Int. J. Comput. Vis.*, vol. 127, no. 4, pp. 363–380, 2019.
[70] D. Zhang, J. Han, L. Yang, and D. Xu, "Spftn: a joint learning framework for localizing and segmenting objects in weakly labeled videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2018.
[71] J. Meng, S. Wu, and W.-S. Zheng, "Weakly supervised person re-identification," in *Proc. Comput. Vis. Pattern Recognit.*, 2019, pp. 760–769.



Zhedong Zheng received the B.S. degree in computer science from Fudan University, China, in 2016. He is currently a Ph.D. student with the School of Computer Science with the University of Technology Sydney, Australia. His research interests include robust learning for image retrieval, generative learning for data augmentation, and unsupervised domain adaptation.

Tao Ruan received the B.E. degree in computer science and technology from Beijing Jiaotong University, Beijing, China, in 2016. He is currently a Ph.D. candidate with the Institute of Information Science, Beijing Jiaotong University, Beijing, China. His research interests include video synopsis and pixel understanding.

Yunchao Wei received the Ph.D. degree from Beijing Jiaotong University, Beijing, China. He was a Postdoctoral Researcher with Beckman Institute, UIUC, from 2017 to 2019. He is currently an Assistant Professor with the Australian Artificial Intelligence Institute, University of Technology Sydney. He is ARC Discovery Early Career Researcher Award (DECRA) Fellow from 2019 to 2021. His current research interests include computer vision and machine learning.

Yi Yang received the Ph.D. degree in computer science from Zhejiang University, Hangzhou, China, in 2010. He is currently a professor with the University of Technology Sydney, Australia. He was a Post-Doctoral Research with the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA. His current research interest includes machine learning and its applications to multimedia content analysis and computer vision.

Tao Mei received the B.E. and Ph.D. degrees from the University of Science and Technology of China, Hefei, China, in 2001 and 2006, respectively. He is a Technical Vice President with JD.COM and the Deputy Managing Director of AI Research of JD.COM, where he also serves as the Director of Computer Vision and Multimedia Lab. Prior to joining JD.COM in 2018, he was a Senior Research Manager with Microsoft Research Asia in Beijing, China. He has authored or coauthored more than 200 publications (with 12 best paper awards) in journals and conferences, 10 book chapters, and edited five books. He holds more than 50 US and international patents (20 granted). He has been an Editorial Board Member for the IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE TRANSACTIONS ON MULTIMEDIA, *ACM Transactions on Multimedia, Pattern Recognition,* etc. He was elected as a fellow of IAPR (2016), a Distinguished Scientist of ACM (2016), and a Distinguished Industry Speaker of IEEE Signal Processing Society (2017), for his contributions to large-scale multimedia analysis and applications.