VehicleNet: Learning Robust Feature Representation for Vehicle Re-identification

Zhedong Zheng[1] Tao Ruan[2] Yunchao Wei[1] Yi Yang[1]
[1]University of Technology Sydney [2] Beijing Jiaotong University

Abstract

Vehicle re-identification (re-id) remains challenging due to significant intra-class variations across different cameras. In this paper, we present our solution to AICity Vehicle Re-id Challenge 2019. The limited training data motivates us to leverage the free data from the web and deploy the two-stage learning strategy. The success of large-scale datasets, i.e.*, ImageNet, inspires us to build a large-scale vehicle dataset called VehicleNet upon the public web data. Specifically, we combine the provided training set with other public vehicle datasets,* i.e.*, VeRi-776, CompCar and VehicleID as VehicleNet. In the first stage, the training set is scaled up about 16 times, from 26,803 to 434,453 images. Despite the bias between different datasets,* e.g.*, illumination and scene, VehicleNet generally provides the common knowledge of the vehicle, benefiting the deeply-learned model in learning the invariant representation towards different viewpoints. In the second stage, we further fine-tune the trained model only on the original training set. The second stage intends to minor the gap between VehicleNet and the original training set. Albeit simple, we achieve mAP 75.60% on the private testing set without extra information,* e.g.*, temporal or spatial annotation of test data.*

1. Introduction

Vehicle re-identification (re-id) is to spot the car of interest in different cameras. It is challenging in the intra-class variants, such as viewpoints, illumination and occlusion. In the realistic scenario, vehicle re-id system demands a robust and discriminative visual representation. Recent years, Convolutional Neural Network (CNN) achieves the state-of-the-art performance in many computer vision tasks including vehicle re-id [11, 15, 22], but CNN is data-hungry and easy to over-fit small datasets. Instead of only using the original training dataset, we first collect free vehicle data from the web. Building upon this, we scale up the number of training images from $26,803$ to $434,453$ as a new dataset called VehicleNet (See Table 1). We train the CNN model to identify different vehicles, and extract features. In the experiment, we show that it is possible to train models with a combination of different datasets. When training the model with more samples, we observe a consistent performance boost, which is consistent with the observation in some recent works [9, 12, 19]. Without explicit vehicle part matching or attribute recognition, the CNN model learns the viewpoint-invariant feature by seeing more vehicles. Albeit simple, the proposed method achieves mAP 75.60% on the private testing set without extra information, *e.g.*, temporal or spatial annotation of test data.

Datasets	# Training Images	Performance	
		Rank@1	mAP
CityFlow [14] †	26,803	73.65	37.65
+ VeRi-776 [11]	+49,357	79.48	43.47
+ CompCar [17]	+136,726	83.37	48.71
+ VehicleID [10]	+221,567	83.37	47.56
VehicleNet	434,453	88.77	57.35

Table 1: 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. We apply SE-ResNeXt101 [7] as the backbone model.

2. Our Approach

2.1. Dataset Analysis

CityFlow [14] 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 at USA City. 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 contain 26,803 images of 255 classes, and the validation query

Datasets	# Cameras	# Images	#IDs
CityFlow [14]	40	56,277	666
VeRi-776 [11]	20	49,357	776
CompCar [17] †	-	136,726	4,701
VehicleID [10] ‡	2	221,567	26,328
PKU-VD1 [16]	-	1,097,649	1,232
PKU-VD2 [16]	-	807,260	1,112
VehicleReID [18]	2	47,123	-
PKU-Vehicle [2]	-	10,000,000	-
Vehicle-1M [4]	2	936,051	55,527
StanfordCars [1]	-	16,185	196

Table 2: Public available datasets for vehicle re-identification. †: 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 [10].

set includes 463 images of the rest 78 classes. We deploy the all original training set as the validation gallery set.

2.2. Extra Datasets

We involve the three public datasets, *i.e.*, VeRi-776 [11], CompCar [17] and VehicleID [10] into training. It results in 434,453 training images of 31,805 classes as **VehicleNet**. Note that the three public datasets are collected in different places with the CityFlow dataset. There are no overlapping images with the validation set or the private test set. We plot the data distribution of all four datasets in Figure 1. **VeRi-776** [11] 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. **CompCar** [17] 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 the 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** [10] consists 2211,567 images of 26,328 vehicles. The vehicle images are collected in two views, *i.e.*, frontal and rear view. Despite the limited viewpoints, the experiment shows that VehicleID also helps the viewpoint-invariant feature learning. **Other Datasets** We also review other public datasets in Table 2. Some datasets contain limited images. Others lack ID annotations. Therefore, we do not use these datasets, which may potentially compromise the feature learning.

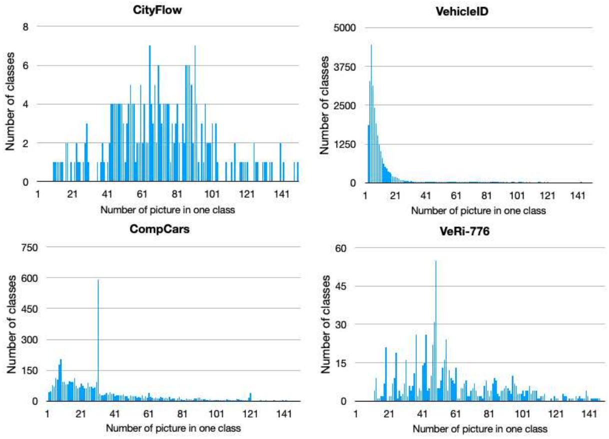

Figure 1: The image distribution per class in the vehicle datasets CityFlow [14], VehicleID [10] , CompCar [17] and VeRi-776 [11]. We observe that the two largest datasets, *i.e.*, VehicleID and CompCars suffer from the limited images per class. Note that there are only a few classes with more than 40 images.

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

Table 3: The Rank@1(%) and mAP (%) accuracy on the validation set with two different sampling methods. Here we use the ResNet-50 backbone.

2.3. Two-Stage Learning

In the first stage, we train the model on VehicleNet illustrated in Section 2.2. We apply the backbone model pre-trained on the ImageNet [13]. The classification layer of the pre-trained model is removed. We add one fully-connected layer of 512 dimensions and one batch normalization layer followed by a new classification layer. The model learns to identify different vehicles from 31, 805 different classes. The cross-entropy loss is applied. As shown in Figure 2 (left), despite a large number of classes, the model could converge within 60 epochs.

In the second stage, the classification layer of the trained model is replaced with the new classifier of 333 classes. We fine-tune the model only upon the original dataset. Attribute to the good initial weights in the first stage, the model converges quickly on the training set (Figure 2 (right)). We, therefore, stop the training early at the 12-th epoch.

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 different sampling policy. The naive method is to sample every image once in

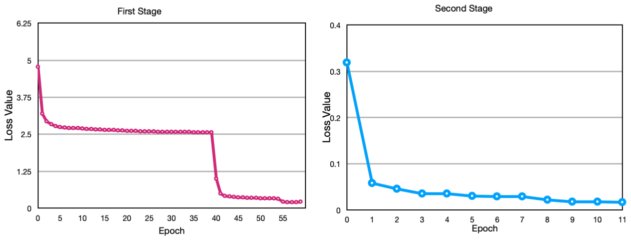

Figure 2: The training losses of the two stages. Due to the large-scale data and classes, the first stage (left) takes more epochs to converge. Attribute to the trained weight of the first stage, the second stage (right) converge early.

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 3, the balanced sampling harms the result. We speculate that the long-tailed data distribution (see Figure 1) makes the balanced sampling have more chance to select the same image in the classes with fewer images. The model, therefore, is prone to over-fit the class with limited samples.

2.4. Post-processing

Several post-processing techniques are leveraged during the inference stage as shown in Figure 3.

Cropped Images. We notice that the challenge provides a relatively loose bounding box. Therefore, we re-detect the vehicle with MaskRCNN [5]. In the submission, the feature is averaged between the original images and cropped images.

Model Ensemble. We adopt a straightforward late fusion strategy, *i.e.*, concatenating the features [20]. Given one image x_i, f_i^j is the extracted feature of the j-th model. The pedestrian descriptor is represented as:$f_i = [|f_i^1|, |f_i^2|, ...|f_i^n|]$. The $|.|$ operator denotes l^2-normalization.

Query Expansion & Re-ranking. We adopt the unsupervised clustering method, *i.e.*, DBSCAN [3] 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 [21] to update the final result.

Camera Verification. We use the camera verification to further remove some hard negative samples. When training, we train one CNN model to recognize the camera that the photo is taken. When testing, we extract the camera-aware features from the trained model and then cluster these features. We applied the assumption that the query image and the target images are taken in different cameras. Given a query image, we remove the images of the same camera cluster from candidate images.

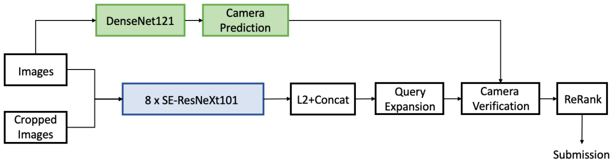

Figure 3: The test pipeline. Given one input image and cropped image, we extract feature from the trained models. We normalize and concatenate the features. Then query expansion and camera verification are applied. Finally, we utilize the re-ranking to retrieve more positive samples.

Backbones	Performance	
	Rank@1(%)	mAP(%)
ResNet-50 [6]	77.97	43.65
DenseNet-121 [8]	83.15	47.17
SE-ResNeXt101 [7]	**83.37**	**48.71**
SENet-154 [7]	81.43	45.14

Table 4: The Rank@1(%) and mAP (%) accuracy with different backbones on the validation set.

3. Experiments

Implementation Details. The images are resized to 384 × 384. We adopt the min-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. We select 8 best models on the validation for further training. In the second stage, we fine-tune the 8 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. 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.

Backbones. We observe that different backbones may lead to different results. As shown in Table 4, SE-ResNeXt101 [7] arrives the best performance in the validation set. We speculate that it is tricky to optimize some large neural networks. We do not achieve a better result with SENet-154 [7], which preforms better then SE-ResNeXt101 on ImageNet.

More Data Matters. As shown in Table 1 , involving more training data consistently improves the result. The model sees more vehicle images taken by different cameras, and learn the viewpoint-invariant features.

Stage I vs. Stage II. We compare the final results of the Stage I and the Stage II on the private test set (see Table 5). The model in Stage II surpasses the one in Stage I about 7%. In the Stage I, the original training set only occupy 6% of VehicleNet. The learned model, therefore, may not be optimal for the original training/test set. Despite the quick training convergence in Stage II, the second stage learning helps

	Performance	
	Rank@1(%)	mAP(%)
Stage I	82.70	68.21
Stage II	87.45	75.60

Table 5: The Rank@1(%) and mAP (%) accuracy with different stages on the private test set. Post-processing methods are leveraged.

Rank	Team Name	mAP(%)
1	Zero_One	85.54
2	UWIPL	79.17
3	ANU	75.89
4	**expensiveGPUs**	**75.60**
5	Traffic Brain	73.02
6	Desire	67.93
7	XINGZHI	60.91
8	UWD_RC	60.78
9	MVM	58.62
10	flyZJ	58.27
	Baseline [14]	32.0

Table 6: Competition results of AICity Vehicle Re-id Challenge. Our result is in **bold**.

to minor the gap between VehicleNet and original training set.

4. Conclusion

In this paper, we present our solution to AICity Challenge. We build a large-scale dataset called VehicleNet with free data from public datasets. The two-stage learning policy and other post-processing techniques are adopted. We arrive at 75.60% mAP on the private testing set. Without extra annotations, our team ranks 4 out of 84 teams (see Table 6). In the future, we will report the result with the help of temporal and spatial information.

References

[1] 3d object representations for fine-grained categorization. In *3DRR*, 2013. 2

[2] Yan Bai, Yihang Lou, Feng Gao, Shiqi Wang, Yuwei Wu, and Ling-Yu Duan. Group-sensitive triplet embedding for vehicle reidentification. *TMM*, 20(9):2385–2399, 2018. 2

[3] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, 1996. 3

[4] Haiyun Guo, Chaoyang Zhao, Zhiwei Liu, Jinqiao Wang, and Hanqing Lu. Learning coarse-to-fine structured feature embedding for vehicle re-identification. In *AAAI*, 2018. 2

[5] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 3

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[7] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018. 1, 3

[8] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2017. 3

[9] Jonathan Krause, Benjamin Sapp, Andrew Howard, Howard Zhou, Alexander Toshev, Tom Duerig, James Philbin, and Li Fei-Fei. The unreasonable effectiveness of noisy data for fine-grained recognition. In *ECCV*, 2016. 1

[10] Hongye Liu, Yonghong Tian, Yaowei Wang, Lu Pang, and Tiejun Huang. Deep relative distance learning: Tell the difference between similar vehicles. In *CVPR*, 2016. 1, 2

[11] Xinchen Liu, Wu Liu, Tao Mei, and Huadong Ma. A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In *ECCV*, 2016. 1, 2

[12] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. In *ECCV*, 2018. 1

[13] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 2

[14] Zheng Tang, Milind Naphade, Ming-Yu Liu, Xiaodong Yang, Stan Birchfield, Shuo Wang, Ratnesh Kumar, David Anastasiu, and Jenq-Neng Hwang. CityFlow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. In *CVPR*, 2019. 1, 2, 4

[15] Zhongdao Wang, Luming Tang, Xihui Liu, Zhuliang Yao, Shuai Yi, Jing Shao, Junjie Yan, Shengjin Wang, Hongsheng Li, and Xiaogang Wang. Orientation invariant feature embedding and spatial temporal regularization for vehicle re-identification. In *ICCV*, 2017. 1

[16] Ke Yan, Yonghong Tian, Yaowei Wang, Wei Zeng, and Tiejun Huang. Exploiting multi-grain ranking constraints for precisely searching visually-similar vehicles. In *ICCV*, 2017. 2

[17] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. A large-scale car dataset for fine-grained categorization and verification. In *CVPR*, 2015. 1, 2

[18] Dominik Zapletal and Adam Herout. Vehicle re-identification for automatic video traffic surveillance. In *CVPR*, 2016. 2

[19] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint discriminative and generative learning for person re-identification. *CVPR*, 2019. 1

[20] Zhedong Zheng, Liang Zheng, and Yi Yang. Pedestrian alignment network for large-scale person re-identification. *TCSVT*, 2018. 3

[21] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. In *CVPR*, 2017. 3

[22] Yi Zhou and Ling Shao. Aware attentive multi-view inference for vehicle re-identification. In *CVPR*, 2018. 1