Going Beyond Real Data: A Robust Visual Representation for Vehicle Re-identification

Zhedong Zheng[1,2‡*] Minyue Jiang[1‡] Zhigang Wang[1] Jian Wang[1] Zechen Bai[1]
Xuanmeng Zhang[1,3] Xin Yu[2] Xiao Tan[1] Yi Yang[2] Shilei Wen[1] Errui Ding[1]
Baidu Inc.[1] University of Technology Sydney[2] Zhejiang University[3]

Abstract

In this report, we present the Baidu-UTS submission to the AICity Challenge in CVPR 2020. This is the winning solution to the vehicle re-identification (re-id) track. We focus on developing a robust vehicle re-id system for real-world scenarios. In particular, we aim to fully leverage the merits of the synthetic data while arming with real images to learn a robust representation for vehicles in different views and illumination conditions. By comprehensively investigating and evaluating various data augmentation approaches and popular strong baselines, we analyze the bottleneck restricting the vehicle re-id performance. Based on our analysis, we therefore design a vehicle re-id method with better data augmentation, training and post-processing strategies. Our proposed method has achieved the ***1st*** *place out of 41 teams, yielding* 84.13% *mAP on the private test set. We hope that our practice could shed light on using synthetic and real data effectively in training deep re-id networks and pave the way for real-world vehicle re-id systems.*

1. Introduction

Powered by artificial intelligence techniques, the Intelligent Transport System (ITS) has drawn increasing interest in both academia and industry [25], as well as improved substantially to the level of applying to real-world problems in modern cities. For example, it optimizes transportation design by estimating of traffic flow characteristics and adaptively adjusting traffic lights to maximize the capability of the transportation. Besides, it also provides comprehensive information about the roads and surrounding environments by detecting vehicles and pedestrians as well as estimating their motions for an automated driving system to generate driving policy.

The perceptual system of an ITS typically consists of following functionalities, including detecting traffic elements, tracking the elements, counting the total number of vehicles in intersections, and estimating poses of vehicles. Vehicle re-identification, a technique of finding the same vehicle in frames captured at different time or even by different cameras, is one of the most critical components in an ITS.

Conventional re-id methods first detect objects independently in frames, followed by a feature extraction step summarizing the appearance feature of the target of interest. Due to the presence of occlusions, noisy detection, different illumination conditions and viewpoint changes, robust appearance feature extraction methods are highly desirable to represent the same objects in different frames. Some approaches resort to object statistical characteristics, such as color histogram or histogram of gradient (HOG), to increase the robustness of feature representations. However, in many challenging situations, hand-crafted statistical features are not capable to represent objects in different views and lighting conditions or with occlusions.

Recently, deep neural network-based re-id approaches [12, 27, 7, 43] have demonstrated superior performance to those hand-crafted feature-based methods. In general, most of them are characterized by a Siamese network and trained over a metric learning objective, such as triplet loss [10], N-pair loss [22] and angulate loss [3]. To be specific, those objectives aim at minimizing the distance of feature representations coming from the same vehicle while pushing the feature representations of different vehicles apart. Moreover, recent extensions take advantage of extra information including vehicle types, colors and vehicle poses to design re-id network architecture and further improve the recognition performance [24]. Additionally, various image generation approaches are introduced to boost the performance of re-id systems. For instance, domain randomization [26] is introduced to generate images by rendering the 3D model of a vehicle with the specified poses and colors. In [39], GAN has been proved to be an effective approach in generating training data for re-id systems.

In this work, we are interested in designing a highly accurate vehicle re-id system in real-world scenarios. Towards this goal, three main problems need to be addressed: (i) how

[*] Work done during a visit at Baidu Inc. [‡] Equal Contribution.

to design a vehicle re-id network effectively and efficiently; (ii) how to incorporate task-specific information to further improve the retrieval performance during testing; (iii) given multiple re-id networks, how to further improve the re-id performance. In this report, we will report our solutions to these key problems and thus provide a strong baseline for the following works.

2. Related Work

The recent advance of vehicle re-identification (re-id) mainly benefit from learned visual deep representation via convolutional neural networks (CNNs) [7, 17, 34, 28]. As reported in [13], effective loss functions, sampling strategies and other training techniques have been proposed in this field to facilitate the CNN learning procedures. For instance, Liu *et al*. [16] fuse the CNN features with the traditional hand-crafted features, yielding robust visual representation. To mine the fine-grained patterns, Wang *et al*. [28] first annotate the key-points of the vehicle images and exploit the part-based vehicle features. Shen *et al*. [21] leverage the prior knowledge that the vehicle usually reappears under cameras during a short time, and apply the spatial-temporal constraints to eliminate the hard-negative samples.

Meanwhile, vehicle re-id methods also take advantage of the experience of other related tasks, *i.e*., person re-identification and face recognition, such as center loss [30], spatial transformer [41] and batch normalization neck [23]. However, real-world vehicle re-identification is still a challenging task due to the large visual appearance changes caused by different cameras, vehicle orientation, illuminations and occlusions.

In order to alleviate the variants and learn the robust vehicle representation, many recent works have explored data generation methods, *i.e*., game engine, and demonstrated the effectiveness of the synthetic data in training re-id networks [39]. Zhou *et al*. [43] propose to synthesize a multi-view feature by transforming a single-view feature against the orientation variation problem, while Yao *et al*. [35] leverage a graphic engine to augment real-world datasets with different orientation and attributes.

Very recently, generative adversarial network (GAN) [6] has been widely used for data generation, which not only transfers the style of image samples from a source domain to a target domain [44, 29, 4], but also generates samples conditioned with the specific attributes [11, 39]. Following this spirit, we also explore different data augmentation approaches and allow the model to "see" more vehicle variants, yielding robust visual representation.

3. Method

We first explore data generation approaches in Section 3.1, following by the representation learning in Section 3.2. When inference, we extract the visual representation from the trained model and conduct the post-processing methods in Section 3.3.

3.1. Synthetic Data

Style Transform. Different from the typical vehicle re-identification dataset, AICity-Flow is composed of both real-world data and synthetic data. We observe that although the identities in synthetic data come from the real world, the synthetic images still present obvious style differences from real images, which is well known as the *domain gap*. To tackle this problem, we utilize the image translation technique. Specifically, a CycleGAN-like framework, *i.e*., UNIT [15], is trained with both real and synthetic data as two different sources. While training, the input images are demanded to be translated across the two sources. After training, we translate all synthetic images in *synthetic* → *real* direction to obtain more realistic samples, which reduces the distribution gap. (see the left part of Figure 1)

Content Manipulation. We note that the above-mentioned style transfer methods do not change the image content. Thus, the generated data is still close to the original inputs in terms of visual appearance, which may limit the learning from the synthetic data. To this end, we also make an attempt to generate new data via content manipulation. DG-Net [39] is a novel framework [1] that can generate samples with different visual appearance, which is particularly effective for the re-id task. It employs two encoders that are respectively responsible for appearance and structure information, while the decoder generates images based on appearance and structure embeddings. In our task, DG-Net is trained on the vehicle re-id dataset provided by the organizer of AICity Challenge. Then the trained model is utilized to generate new identities. As shown in the right part of Figure 1, given images of two identities in different colors, DG-Net is to generate new images with the target appearance. To avoid ambiguation caused by similar identities as well as the failure cases due to the low-resolution images, we apply the generation on a high-resolution subset of the dataset. Furthermore, to force the generated data to possess consistent appearance, we select only one target image to provide appearance embedding for the whole source images. The generated data is only used in fine-tuning stage, we will provide more details in Section 3.2.

Copy & Paste. Besides, we also explore the straightforward method, *i.e*., copy and paste, to augment training data and let the model "see" more background variants. We generate new samples by combining the foreground of the

[1] https://github.com/NVlabs/DG-Net

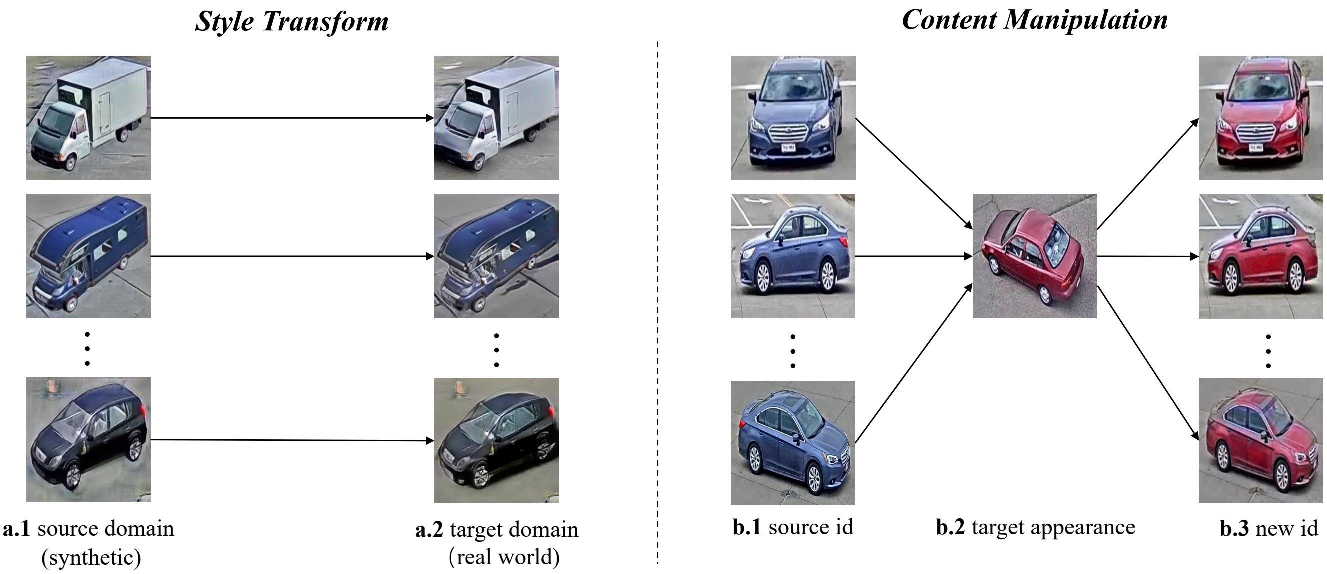

Figure 1. **Style Transform**. On the left side, to meet real-world distribution, **a.1** images in the synthetic domain are transformed into real-world style as in **a.2**. **Content Manipulation**. On the right side, **b.1** is the images of one identity selected from the original dataset. We set **b.2** as the target appearance image, and apply DG-Net to generate the new samples in **b.3** that possess both structure of **b.1** and appearance of **b.2**. We regard the generated images with the target appearance as one new vehicle category, and involve them in the training.

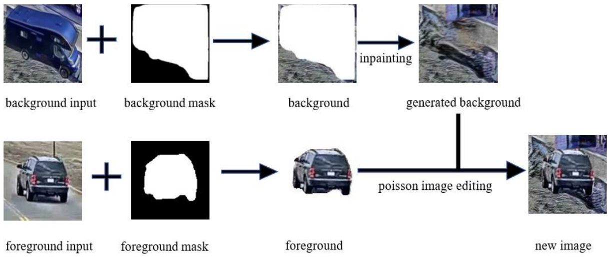

Figure 2. The Copy & Paste procedure. Given one foreground input and one background input, we first apply MaskRcCNN [8] to obtain the vehicle mask. Then we use DeepFill v2 to conduct image inpainting on the background image. Finally, we deploy the seamless image cloning to "paste" the vehicle region onto the background image.

real images with the background of the synthetic images. In particular, for the foreground extraction, we segment the vehicle from the real image via instance segmentation approach, *i.e.*, MaskRCNN [8]. For the background, we apply DeepFill v2 [36] to conduct the image inpainting on the blank area where the foreground is removed. The total procedure is shown in Figure 2. Finally, we apply the seamless image cloning to fuse the foreground and the background images.

3.2. Representation Learning

Network Structure. Following existing re-id works [40, 37], we deploy the state-of-the-art networks pre-trained on ImageNet [2] as the backbone module, including ResNeXt101[32], ResNeXt101_32x8l_wsl [19] and ResNet50_IBN_a[33]. Specifically, we deploy open-source network structure variants as follows:

- The vanilla re-id baseline [2] replaces the original classification layer for ImageNet with one new classifier module. The new classifier module contains one fully-connected layer $fc1$, one batch normalization layer and one fully-connected layer $fc2$. The first $fc1$ layer is to compress the learned feature to 512 dimension, when the second $fc2$ layer could be viewed as a linear classifier to output the category prediction. When inference, we extract the 512-dim feature before the $fc2$ layer as the visual representation.

- Moreover, we also explore another sophisticated re-id network architecture [3], which fuses multi-scale information to enhance the vehicle representation. Figure 3 briefly illustrates the architecture of this network. Specifically, the activations of the last two block of the ResNet backbone, *i.e.*, Block3 and Block 4, are employed. We denote the two features as $X3$ and $X4$, respectively. Global average pooling (GAP) and global max pooling (GMP) are used to obtain the global representations. Besides, adaptive average pooling (AAP) and adaptive max pooling (AMP) with output size 2×2 are conducted on $X4$ to get the local

[2] https://github.com/layumi/Person_reID_baseline_pytorch

[3] https://github.com/douzi0248/Re-ID

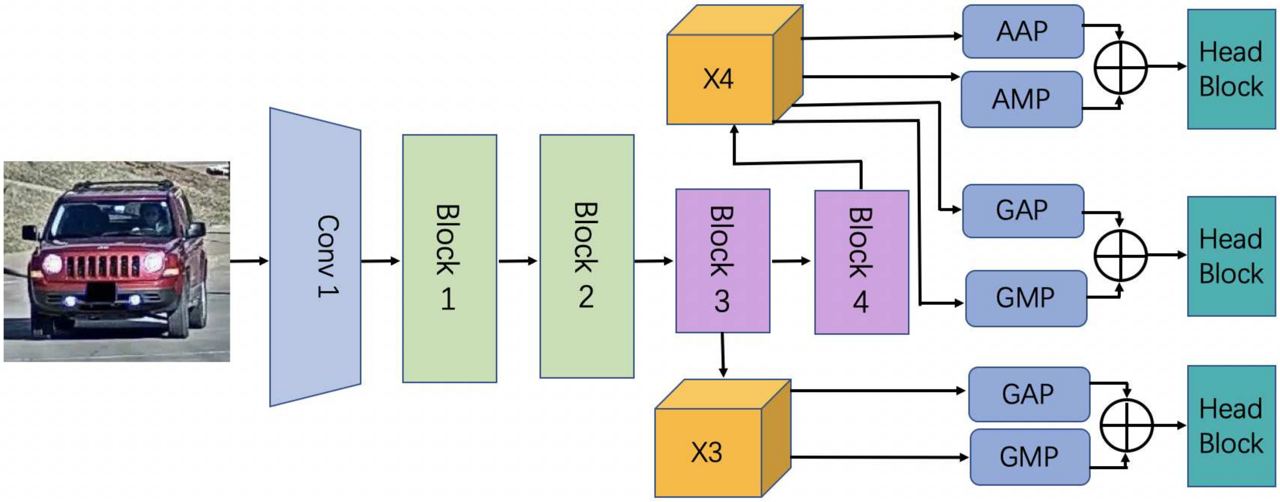

Figure 3. The network architecture of one typical model used in this work. GAP and GMP indicate global average pooling and global max pooling, respectively, while AAP and AMP denote adaptive average pooling and adaptive max pooling. Head Block contains one Batch normalization (BN) layer, leaky-relu function, one convolutional layer and one BN layer, followed by the fully-connected (fc) layer.

representations. $X3\_g\_avg$ indicates the global average pooling feature of $X3$, while $X4\_a\_max$ indicates the adaptive max pooling feature of $X4$. Similarly, we also obtain $X3\_g\_max$, $X4\_g\_avg$, $X4\_g\_max$ and $X4\_a\_avg$. All aforementioned output features are supervised by the ranking loss to pull samples of the same identity closer and push samples of different identities far away from each other in the feature space. $X3\_g\_avg$ and $X3\_g\_max$ are further fed to the head block, so are $X4\_g\_avg$ and $X4\_g\_max$, $X4\_a\_avg$ and $X4\_a\_max$. The head block contains a batch normalization (BN) layer, a leaky-relu layer, a convolutional (Conv) layer, another batch normalization layer and a fully connected (fc) layer to predict the vehicle identity. The cross-entropy loss is utilized to punish the incorrect prediction.

Optimization Functions. We deploy the two widely-adapted objectives, *i.e.*, the cross-entropy loss and the ranking loss to optimize the model. We denote N as the number of vehicle identities in the dataset. Given the input image x and the corresponding label y, the cross-entropy loss is to penalize the wrong category prediction, which could be formulated as:

$$loss_{ce} = -\sum_{i=1}^{N} p_i log(\hat{p_i}), \tag{1}$$

where p_i is the ground truth label of the input sample x. $p_i = 1$ if i equals to the ground truth label y, else $p_i = 0$. $\hat{p_i}$ is the predicted probability.

In contrast, the ranking loss focuses on optimizing the distance between the training samples. The intuition is to pull the feature of the positive pairs close, while pushing the features of the samples from different vehicle identities away from each other by a large margin. Given the triplet $\{x_a, x_p, x_n\}$, x_a and x_p are the samples of the same vehicle, while x_a and x_n are of different identities. The ranking loss could be formulated as:

$$loss_{ranking} = [D_{ap} - D_{an} + m]_+, \tag{2}$$

where $D_{ap} = ||f(x_a) - f(x_p)||$, $D_{an} = ||f(x_a) - f(x_n)||$, m is the margin and $[\cdot]_+$ denotes the hinge function $max(0, \cdot)$. $|| \cdot ||$ denotes the L2-norm.

Negative Mining. To enhance the discriminability of the learned model, we apply an off-line negative example mining step to fine-tune the model. It contains two stages, *i.e.*, negative mining and regular training. In the negative mining stage, we randomly sample 50% images from the mini-batch, then the most similar negative pairs are selected to comprise the hard-negative training triplet. As a result, we could obtain the challenging training samples, which is close to the decision boundary to help the model learning. The second stage is to train the model using the ranking loss as usual.

Auxiliary Information Learning. We find that the vehicle re-id model is easily confused by different samples with similar orientation. To overcome this drawback, we employ a direction classification model to predict the orientation of each vehicle and depress some vehicle pairs according to their orientation similarity in the post-processing

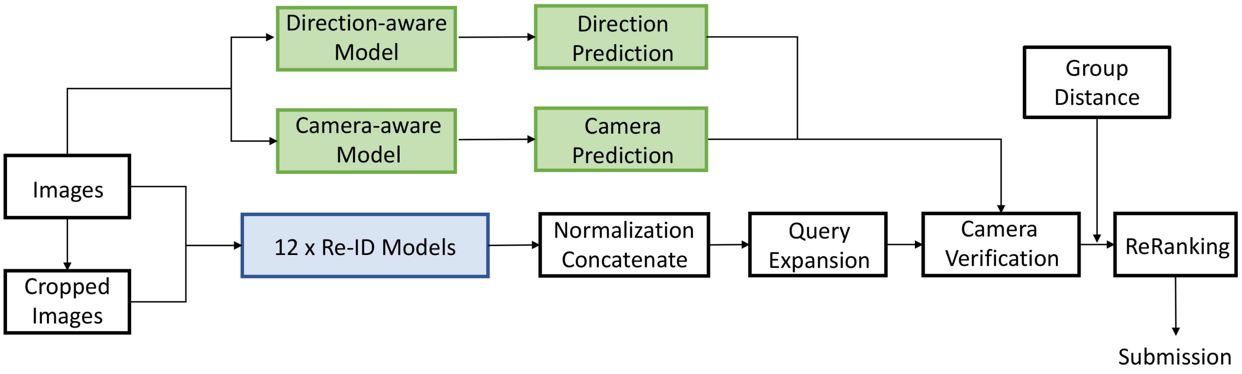

Figure 4. The inference pipeline. Given one input image and cropped image, we extract features from the trained models. We normalize and concatenate the features. Then query expansion and camera verification are applied. Finally, we utilize the group distance and re-ranking to retrieve more positive samples.

stage. The network and corresponding annotation-extended training set have already been released [4]in AICity Challenge 2019. The direction classification model is simple yet effective, which follows a standard classification network architecture. ResNet50 [9] is selected as the backbone following by a global average pooling layer. The dropout operation is employed to avoid overfitting. Then, a convolution layer and a batch normalization layer are stacked to reduce the feature dimension. Finally, a fully connected layer maps the feature to the number of predefined orientations. In the testing stage, each image is passed through the direction classification model to get the orientation probability vectors. Finally, we use the dot product of two orientation probability vectors to represent the orientation similarity of a pair of images. Besides, we also train a camera-aware model to predict the camera where the vehicle image is captured. The camera-aware model is combined with the direction-aware model to enable the camera verification in the post-processing pipeline. More details are provided in Section 3.3.

Implementation Details. We train the model using stochastic gradient descent (SGD) with momentum 0.9 based on the paddlepaddle framework [5]. The base learning rate is set to 0.001. We adopt the cosine strategy to decay the learning rate [18]:

$$lr = base\_lr \times 0.1^{\left\lfloor \frac{Epoch}{30} \right\rfloor} \times cos(\pi * (Epoch\%30)), \quad (3)$$

The input images are resized to $\{384, 400, 416\}$ to train different models for further ensemble. We also apply the common data augmentation, including random flip, scale jittering, and the learned augmentation policy on ImageNet [1].

We use detectron2 [31] as the instance segmentation tool to extract vehicle mask from the image. Taking X101-FPN as the backbone, the model is trained on COCO dataset train2017 [14]. Besides, seamless image cloning [20] is used to copy the vehicle region from a foreground image onto a background image subject to removal of visual seams.

Fine-tuning Model. In order to force the model to better meet the real-world application, after the standard training procedure, the model is further fine-tuned upon the real-world data with a smaller learning rate [38]. Since the category number of the realistic data is less than the number of realistic data with the synthetic data, we replace the classifier of the trained model with a new classifier layer. Thus, during the fine-tuning stage, we adopt a warm-up policy that first optimizes the new classifier layer while fixing the backbone network. After that all parameters in the network are fine-tuned. Apart from data from the original dataset, we also utilize the DG-Net [39] generated data as well as the cropped data to fine-tune diverse models for the model ensemble in the post-processing.

3.3. Post-Processing

Furthermore, we also deploy several post-processing techniques to facilitate the final retrieval results (see Figure 4). Specifically, the approaches contain image alignment, model ensemble, query expansion, re-ranking, camera verification and group distance.

Image Alignment. We notice that the Challenge dataset provides a relatively loose bounding box, which may introduce the extra background [41]. Thus, we re-detect the vehicle with the state-of-the-art MaskRCNN [8]. To arrive the final result, the vehicle representation is averaged between the original images and cropped images to obtain more robust vehicle representations.

Model Ensemble. We adopt the similar policy in [38] to conduct the feature-level ensemble. In particular, we con-

[4] https://github.com/wzgwzg/AICity
[5] https://www.paddlepaddle.org.cn/

catenate the normalized features from 12 different models as the final visual representation.

Query Expansion & Re-ranking. We adopt the unsupervised clustering method, *i.e.*, DBSCAN [5] to find the most similar samples. The query feature is updated to the mean feature of the other queries in the same cluster. We notice that low-resolution images may compromise the feature discriminability. Thus, we do not involve the feature of low-resolution image into the calculation of the mean feature. Furthermore, we adopt the re-ranking method [42] 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 several camera-aware CNN models to recognize the camera from which the vehicle image is taken. When testing, we extract camera predictions and 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 reduce the similarity of the images of the same camera prediction or the same camera cluster center from candidate images (gallery). Besides, as shown in [25], we observe that the cameras #6, #7, #8, #9 are located at a crossroad, and the direction of the vehicles are mostly different. **We note that the training data only contains real images from the Scenarios 1, 3, and 4 without cameras #6, #7, #8, #9. Thus, while inference, we assume that the images with low camera prediction confidence are from #6, #7, #8, #9. We do not use any extra camera annotation of the test data.** Based on this assumption, we further add one direction constraints that the query image and the target images also should have different direction predictions.

Group Distance. The tracklet information of vehicles is provided in AICity Challenge, which is close to the realistic scenario. In the real-world application, the tracklet could be obtained via vehicle detection and tracking algorithm under the same camera. To leverage such information, we introduce two assumptions as follows: 1) The image from the same tracklet is of the same vehicle, and could share the visual representation to enhance the representation scalability of a single image; 2) Different tracklets under the same camera are of different vehicles. Based on the first assumption, we adopt the gallery expansion, which updates the gallery feature to the mean feature of the other images in the same tracklet. In contrast, based on the second assumption, we introduce an aggressive strategy to reduce the similarity of hard negative samples. Given the high-confidence retrieved image from the camera C, we reduce the similarity score of different tracklets from the same camera C.

Table 1. Competition results of AICity Vehicle Re-id Challenge. Our result is in **bold**.

Rank	Team Name	mAP(%)
1	**Baidu-UTS**	**84.13**
2	RuiYanAI	78.10
3	DMT	73.22
4	IOSB-VeRi	68.99
5	BestImage	66.84
	Baseline [25]	32.0

Table 2. Ablation Study. The Rank@1(%) and mAP (%) accuracy with / without synthetic training data.

	Performance	
	Rank@1(%)	mAP(%)
without Synthetic Data	79.78	43.87
with Synthetic	80.86	46.90

Table 3. Ablation Study. Effect of different post-processing techniques on the validation set.

Method	Performance					
with Alignment?		✓	✓	✓	✓	✓
Query Expansion?			✓	✓	✓	✓
Camera Verification?				✓	✓	✓
Group Distance?					✓	✓
Re-ranking?						✓
mAP (%)	46.90	47.66	49.06	50.07	51.58	61.26

4. Experiment

4.1. Dataset Analysis

This challenge is based on CityFlow dataset [25], which consists of $36,935$ training images of 333 vehicles. The private test set contains $1,052$ query images and $18,290$ gallery images. This year, the organizer also provides the synthetic data from [35], including $192,150$ images of $1,362$ vehicles. The validation set is not provided, so we split one validation set from the training set to conduct the ablation studies of the important components. We follow the split in [38], which leaves out the last 78 vehicle ID as the validation set.

4.2. Quantitative Results

Comparison with Other Teams. As shown in Table 1, the proposed method has achieved the state-of-the-art mAP accuracy, *i.e.*, 84.13%, which is superior to the second-best team by a large margin and verifies the effectiveness of the proposed re-id method.

Effect of the Synthetic Data. First of all, we evaluate the

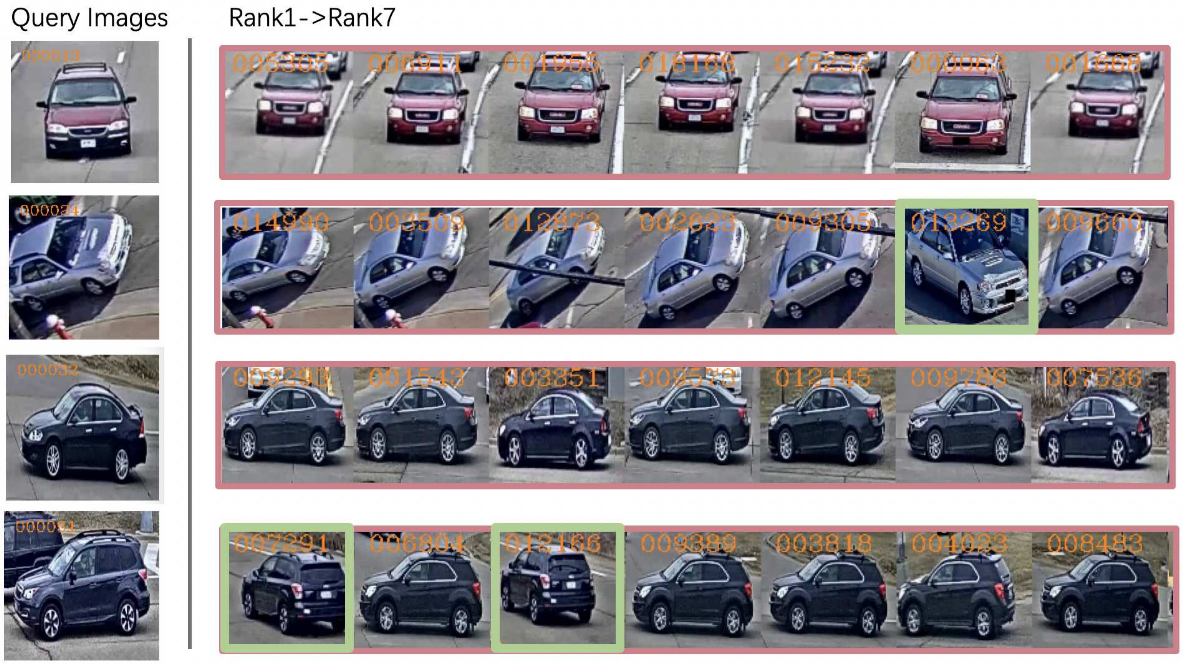

Figure 5. Visualization of hard-negatives before post-processing. The first column shows the selected query images captured by different cameras, and each row shows the top 7 gallery images retrieved from left to right according to the similarity score. The images in green boxes are true positives, while the images in red boxes are false positives.

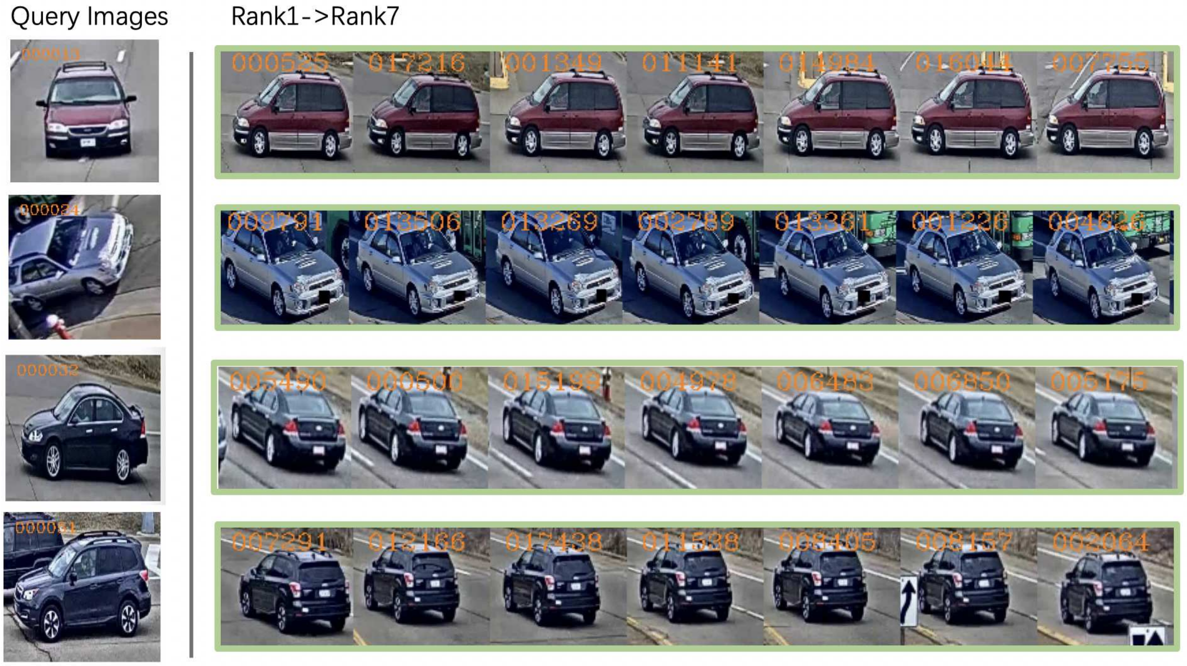

Figure 6. Visualization of the final retrieval results. The first column shows the query images captured by different cameras, and each row shows the top 7 gallery images retrieved from left to right according to the similarity score. The images in green boxes are true positives, while the images in red boxes are false positives. We observes that the post-processing techniques could successfully eliminate the hard-negatives with similar directions in Figure 5.

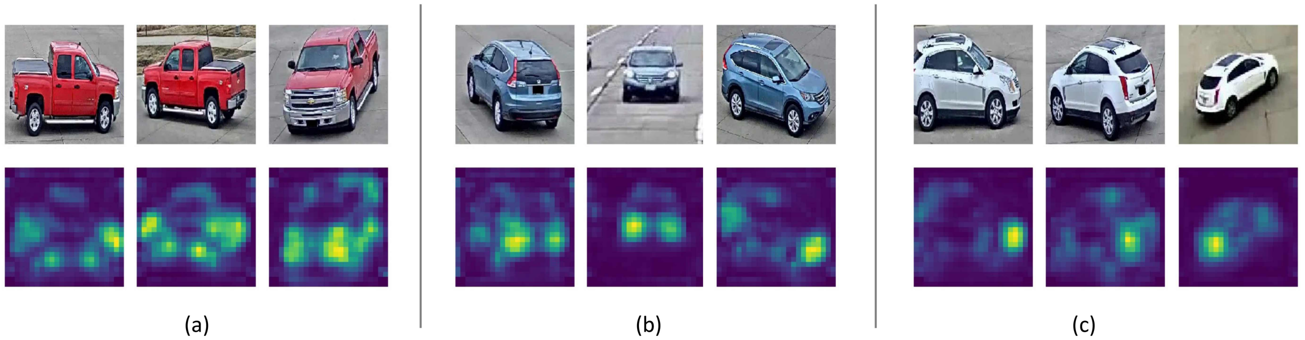

Figure 7. Visualization of feature maps. Following [40], we visualize the activation map before the final pooling layer. Similar to the human, the learned model has a strong attention to the discriminative parts, such as the car light and the tire type.

effectiveness of the synthetic data on the validation set. We have trained and evaluated the model with and without synthetic data respectively. As shown in Table 2, the model trained with synthetic data has achieved 80.86% Rank@1 and 46.90% mAP, which is superior to the model trained without synthetic data.

Effect of the Post-processing. Furthermore, we evaluate the proposed post-processing methods on the validation set. We gradually add the post-processing techniques (see Table 3), yielding the superior performance. On the validation set, we improve the vanilla baseline from 46.90% mAP to 61.26 mAP after re-ranking.

4.3. Qualitative Results

Visualization of Retrieval Results. We visualize the ranking lists without or with post-processing respectively (see Figure 5 and Figure 6). We select some hard-negative samples in the test set. From the changes of the ranking list, we observe that although models are already very powerful, there still remain lots of queries, which cannot find the right matches due to the hard-negatives with similar poses. With the help of post-processing, many cases can be rectified.

Visualization of Feature Maps. We further visualize the heatmap of the learned model (see Figure 7). Given the input image, we follow the visualization approach in [40] to calculate the sum of the activation before the final pooling layer. We observe that the learned model takes more attention to the discriminative parts, such as the car lights, which is aligned with the human experience. Thus, the vehicle representation is robust to the visual appearance changes due to the large viewpoint variants.

5. Conclusion

In this paper, we develop a robust vehicle re-id system for vehicle re-identification, yielding the first place in the re-id track of AICity Challenge 2020. We verify the effectiveness of the synthetic data in learning the robust visual representation and explore different popular baselines and generation models in the context of vehicle representation learning. In the future, we will continue to study the 3D vehicle models and other relevant techniques to facilitate vehicle re-id in real-world applications.

References

[1] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *CVPR*, 2019. 5

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 3

[3] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, 2019. 1

[4] Weijian Deng, Liang Zheng, Guoliang Kang, Yi Yang, Qixiang Ye, and Jianbin Jiao. Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person reidentification. In *CVPR*, 2018. 2

[5] 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. 6

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014. 2

[7] 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. 1, 2

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

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[10] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv:1703.07737*, 2017. 1

[11] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, 2018. 2

[12] Minyue Jiang, Yuan Yuan, and Qi Wang. Self-attention learning for person re-identification. In *BMVC*, 2018. 1

[13] Ratnesh Kumar, Edwin Weill, Farzin Aghdasi, and Parthsarathy Sriram. Vehicle re-identification: an efficient baseline using triplet embedding. *arXiv:1901.01015*, 2019. 2

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5

[15] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *NeurIPS*, 2017. 2

[16] Weiyang Liu, Yandong Wen, Zhiding Yu, and Meng Yang. Large-margin softmax loss for convolutional neural networks. In *ICML*, 2016. 2

[17] 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. 2

[18] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv:1608.03983*, 2016. 5

[19] 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. *arXiv:1901.01015*, 2018. 3

[20] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. *ACM Transactions on Graphics*, 22(3):313–318, 2003. 5

[21] Yantao Shen, Tong Xiao, Hongsheng Li, Shuai Yi, and Xiaogang Wang. Learning deep neural networks for vehicle re-id with visual-spatio-temporal path proposals. In *ICCV*, 2017. 2

[22] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in neural information processing systems*, 2016. 1

[23] Yifan Sun, Liang Zheng, Weijian Deng, and Shengjin Wang. SVDNet for pedestrian retrieval. In *ICCV*, 2017. 2

[24] Zheng Tang, Milind Naphade, Stan Birchfield, Jonathan Tremblay, William Hodge, Ratnesh Kumar, Shuo Wang, and Xiaodong Yang. Pamtri: Pose-aware multi-task learning for vehicle re-identification using highly randomized synthetic data. In *ICCV*, 2019. 1

[25] 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, 6

[26] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 23–30, 2017. 1

[27] Guanshuo Wang, Yufeng Yuan, Xiong Chen, Jiwei Li, and Xi Zhou. Learning discriminative features with multiple granularities for person re-identification. In *ACM MM*, 2018. 1

[28] 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. 2

[29] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. Person transfer GAN to bridge domain gap for person re-identification. In *CVPR*, 2018. 2

[30] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*. Springer, 2016. 2

[31] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019. 5

[32] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017. 3

[33] Jianping Shi Xingang Pan, Ping Luo and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *ECCV*, 2018. 3

[34] 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

[35] Yue Yao, Liang Zheng, Xiaodong Yang, Milind Naphade, and Tom Gedeon. Simulating content consistent vehicle datasets with attribute descent. *arXiv:1912.08855*, 2019. 2, 6

[36] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. *arXiv:1806.03589*, 2018. 3

[37] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jiahao Bu, and Qi Tian. Person re-identification meets image search. *arXiv:1502.02171*, 2015. 3

[38] Zhedong Zheng, Tao Ruan, Yunchao Wei, and Yi Yang. Vehiclenet: Learning robust feature representation for vehicle re-identification. In *CVPR Workshops*, pages 1–4, 2019. 5, 6

[39] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint discriminative and generative learning for person re-identification. In *CVPR*, 2019. 1, 2, 5

[40] Zhedong Zheng, Liang Zheng, and Yi Yang. A discriminatively learned CNN embedding for person reidentification. *ACM TOMM*, 2017. 3, 8

[41] Zhedong Zheng, Liang Zheng, and Yi Yang. Pedestrian alignment network for large-scale person re-identification. *TCSVT*, 2018. 2, 5

[42] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. In *CVPR*, 2017. 6

[43] Y Zhou, L Shao, and A Dhabi. Viewpoint-aware attentive multi-view inference for vehicle re-identification. In *CVPR*, 2018. 1, 2

[44] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *NeurIPS*, 2017. 2