An Empirical Study of Vehicle Re-Identification on the AI City Challenge

Hao Luo[1], Weihua Chen[1], Xianzhe Xu[1], Jianyang Gu[1], Yuqi Zhang[1], Chong Liu[1]
Yiqi Jiang[1], Shuting He[1], Fan Wang[1], Hao Li[1]
Machine Intelligence Technology Lab, Alibaba Group
michuan.lh@alibaba-inc.com

Abstract

This paper introduces our solution for the Track2 in AI City Challenge 2021 (AICITY21). The Track2 is a vehicle re-identification (ReID) task with both the real-world data and synthetic data. We mainly focus on four points, i.e. *training data, unsupervised domain-adaptive (UDA) training, post-processing, model ensembling in this challenge. (1) Both cropping training data and using synthetic data can help the model learn more discriminative features. (2) Since there is a new scenario in the test set that dose not appear in the training set, UDA methods perform well in the challenge. (3) Post-processing techniques including re-ranking, image-to-track retrieval, inter-camera fusion, etc, significantly improve final performance. (4) We ensemble CNN-based models and transformer-based models which provide different representation diversity. With aforementioned techniques, our method finally achieves 0.7445 mAP score, yielding the first place in the competition. Codes are available at* https://github.com/michuanhaohao/AICITY2021_Track2_DMT.

Figure 1. Some examples of the real-world and synthetic data.

1. Introduction

Vehicle ReID is an important computer-vision task which aims to identify the target vehicle in images or videos across different cameras, especially without knowing the license plate information. Vehicle ReID is important for intelligent transportation systems (ITS) of the smart city. For instance, the technology can track the trajectory of the target vehicle and detect traffic anomalies. Recently, most of works have been based on deep learning methods in vehicle ReID, and these methods have achieved great performance in vehicle ReID.

As shown in Figure 1, Track2 provides a real-world dataset CityFlow-V2 [22] and a synthetic dataset VehicleX [27] for model training. However, CityFlow-V2 only contains 52,712 images, which is not enough to train a robust model. Therefore, the first challenge is how to overcome the lack of real data. For the real data, inaccurate bounding boxes introduce noise in the image. We re-detect real-world images according to their heatmaps and add them into training data. Adding synthetic images into training data also can improve the performance. In addition, we also try to use SPGAN [2] to transfer synthetic data into 'real-world' data.

The second challenge observed is that a new scenario appears in the test set, *i.e*. there exists domain bias between the training and test sets. Unsupervised domain-adaptive (UDA) methods are suitable to address the problem. In the second stage, We perform UDA methods to automatically generate pseudo labels on the testing data, which are used to fine-tune models. To improve the quality of pseudo-labels, we consider tracklet information and camera bias between images in the clustering process. The UDA training can significantly improve the model performance on the test set.

Thirdly, several post-processing techniques are leveraged during the inference stage. For example, a camera

model and a orientation model are trained to reduce the camera and orientation bias between images, respectively. Since the tracklet information is provided, we integrate all features of a tracklet to obtain a more robust feature for the tracklet in the inference stage. Camera verification [30] based on a camera-connection constraint is also introduced to remove negative images from candidate images. In addition, some widely used methods including the re-ranking [31] and inter-camera fusion also improve a lot of accuracy.

Finally, we ensemble multiple models to further improve scores on the leaderboard. It is note that we have try both CNN-based methods and transformer-based methods (*i.e*. TransReID [10]). To our best knowledge, it is the first time to study pure-transformer models on the AI City Challenge. Our experience shows that TransReID can provide diversity that is different from CNN-based models. Therefore, we ensemble CNN-based methods and TransReID to finally achieve 0.7445 mAP score, yielding the first place on the leaderboard.

Our contributions can be summarized as follow:

- We conduct a empirical study of vehicle ReID on AICITY21. We have tried many methods that have been verified in the ReID field in the past few years.
- We observe domain bias between the training and testing data and introduce the UDA training to address the challenge.
- We try transformer-based models on the AI City Challenge for the first time. The experimental results show that transformer-based models can provide representation diversity different from CNN-based models.
- We achieves 0.7445 mAP socre without external data, yield the first place in the competition.

2. Related Works

We introduce deep ReID and some works of AICITY2020 in this section.

2.1. Deep ReID

Re-identification (ReID) is widely studied in the field of computer vision. This task possesses various important applications. Most existing ReID methods based on deep learning. Recently, CNN-based features have achieved great progress on both person ReID and vehicle ReID. Person ReID provides a lot of insights for vehicle ReID. Luo *et al*. [16, 17] proposed a strong baseline [16, 17] in person ReID, which also performers well in vehicle ReID. For vehicle ReID, Liu *et al.* [15] introduced a pipeline that uses deep relative distance learning (DRDL) to project vehicle images into an Euclidean space, where the distance can directly measure the similarity of two vehicle images. Shen *et al.* [21] proposed a two-stage framework that incorporates complex spatial-temporal information of vehicles to effectively regularize ReID results. Zhou *et al.* [32] designed a viewpoint-aware attentive multi-view inference (VAMI) model that only requires visual information to solve multi-view vehicle ReID problems. While He *et al.* [7] proposed a simple yet efficient part-regularized discriminative feature-preserving method, which enhances the perceptive capability of subtle discrepancies, and reported promising improvement. Some works [33, 20] also studied discriminative part-level features for better performance. Some works [25, 14, 13] in vehicle ReID had utilized vehicle key points to learn local region features. Several recent works [7, 24, 6, 23] in vehicle ReID had stated that specific parts such as windscreen, lights and vehicle brand tend to have much discriminative information. In recently, a pure-transformer method called as TransReID [10] has shown that transformer-based methods achieves better performance than CNN-based methods on some vehicle ReID benchmarks.

2.2. AICITY20

Since AICITY21 is updated from AI CITY Challenge 2020 (AICITY20), some methods of AICITY20 are helpful for our solution. The organizers outlined the methods of leading teams in [19]. Zheng *et al*. [29] trained real data with synthetic data by applying style transformation and content manipulation. Zhu *et al*. [34] proposed an approach named VOC-ReID, taking the triplet vehicle-orientation-camera as a whole and reforming background/shape similarity as camera/orientation re-identification. He *et al*. [9] proposed the Identity Mining method to automatically generate pseudo labels on the test data to expand the training set. Some other works also studied on loss functions, model structures, post-processing strategies, etc.

3. Our Method

Our solution includes three parts, *i.e*. baseline training, UDA training, and post-processing.

3.1. Stage1: Baseline Training

In this section, we will introduce models and training data used to train robust baseline models.

3.1.1 Baseline Model

Baseline model is important for the final ranking. In track2, we use a CNN-based baseline [16, 17] shown in Figure 2 and a transformer-based baseline [10] shown in Figure 3.

Similar with many past works on the AI CITY Challenge, we choose Bag of Tricks (BoT) as the CNN-based baseline. The version of BoT is the one proposed by ours [9] in the 2020 AI City Challenge. In the modified version,

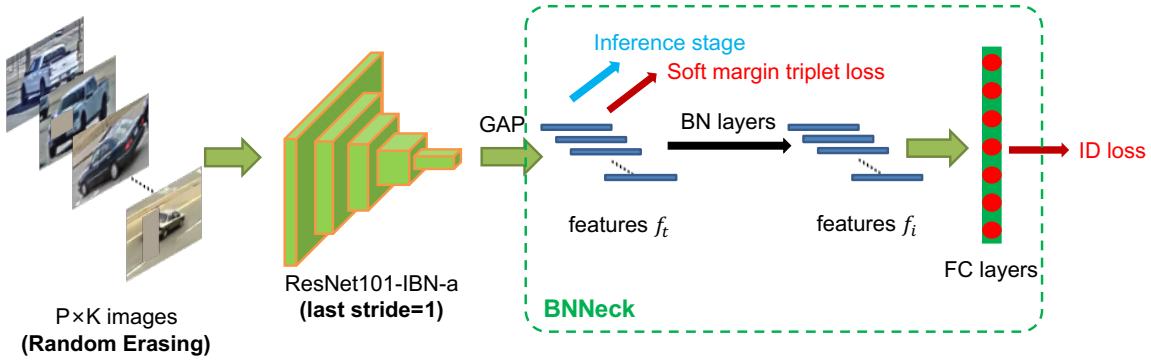

Figure 2. The framework of our CNN-based baseline BoT [9].

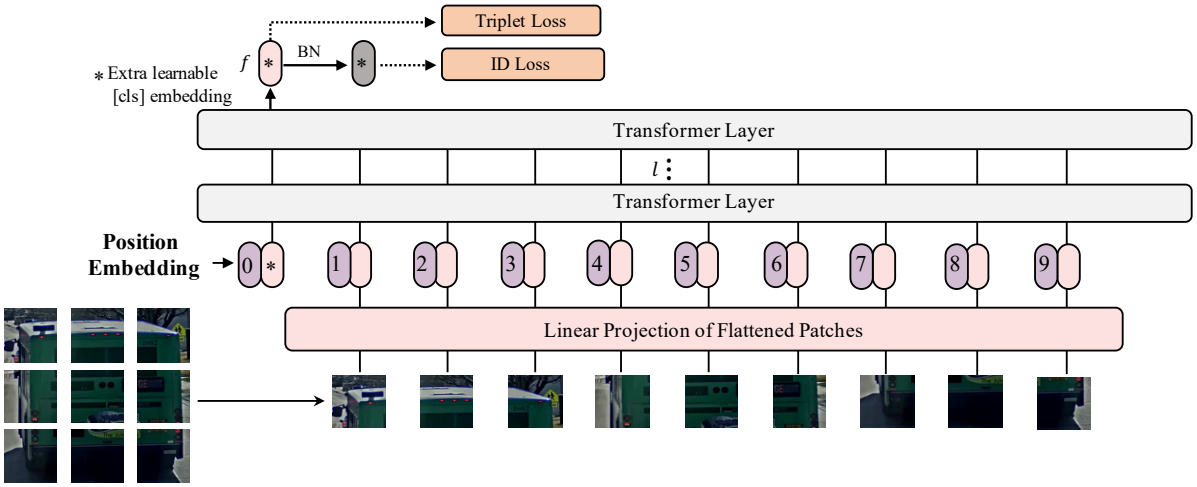

Figure 3. The framework of our transformer-based baseline TransReID [10].

both label smoothing and center loss are removed. In addition, the triplet loss is the soft-margin version as follow:

$$\mathcal{L}_{Tri} = \log\left[1 + \exp(||f_a - f_p||_2^2 - ||f_a - f_n||_2^2 + m)\right] \quad (1)$$

We also use a transformer-based baseline called TransReID [10] that is the first pure-transform model in the field of ReID. To simplify our codes, we remove the local branch (the JPM branch) and only use the global feature of TransReID. But our experiments show that the JPM branch still performs well on CityFlow-V2. Since the dataset does not provide accurate orientation labels, we also ignore the SIE module.

3.1.2 Training data

The Track2 provides two training sets (CityFlow-V2 and VehicleX). CityFlow-V2 is not a large-scale dataset to train robust ReID models. Therefore, a challenge is to overcome the lack of training data. VehicleX additionally provides 192,150 images to expand training data. However, all these images are synthesis images generated by a 3D engine. The domain bias between CityFlow-V2 and VehicleX should be considered. To address these challenges, we introduce our training sets as follow.

CityFlow-V2 (CFV2). The CityFlow-V2 dataset[18, 22] is a real-world dataset that is captured by 46 cameras in real-world traffic environment. It totally includes 85,058 images of 880 vehicles. 52,717 images of 440 vehicles are used for training. The remaining 31,238 images of 440 vehicles are for testing. It is note that the training set is captured by 40 cameras. In the test set, part of images are captured by 6 new cameras that do not appear in the training set.

CityFlow-V2-CROP (CFV2-C). To overcome the lack of real-world data, we use the weakly supervised detection [34] to increase the training data. We train an initial vehicle ReID model to get heatmap response of each image and setting a threshold to get the bounding box larger than it. Then, we get a cropped copy of both training and test set. The modified dataset is called as CityFlow-V2-CROP (CFV2-C) in this paper. After weakly supervised detection, the real-world data becomes doubled. Some examples are shown in Figure 4.

(a) Original images (b) Heatmaps (b) Cropped images

Figure 4. Some examples of cropped images generated by weakly supervised detection.

Figure 5. Some examples of 'real-world' generated by SPGAN. The first and second rows show original synthetic images and generated images, respectively.

VehicleX (VeX). The VehicleX dataset is a synthetic dataset generated by a publicly available 3D engine VehicleX [27]. The dataset only provides training set that contains 192,150 images of 1,362 vehicles in total. In addition, the attribute labels, such as car colors, car types, orientation labels are also annotated. In Track2, the synthetic data can be used for the model training or transfer learning. However, there exists domain bias between the real-world and synthetic data.

VehicleX-SPGAN (VeX-S). We use SPGAN [2] to transfer synthetic images to 'real-world' images to reduce the domain bias. The generated images construct a new dataset called as VehicleX-SPGAN (VeX-S) in this paper. The number of images in VehicleX-SPGAN is same with the number of images in VehicleX.

3.2. Stage2: UDA Training

As we mentioned above, a new scenario appears in the test set, which results in the domain bias between the training and test sets. It is a typical UDA ReID task. In the second stage, we use clustering algorithms to generate pseudo labels on the testing data and then fine-tune baseline models.

For an image I that belongs to the C-th camera and the T-th tracklet in the test set, its global feature is denoted as g_I. Then we compute the average feature of all images captured by the C-th camera and denote the average feature as $\overline{g}_C$. To reduce the camera bias between cross-camera images, the single-frame feature f_I of I is computed as follow:

$$f_I = g_I - \alpha \overline{g}_C \quad , \tag{2}$$

where α is the balance weight between g_I and $\overline{g}_C$. When we average features of all IDs in the c-th camera, the ID information is ignored. Therefore, $\overline{g}_C$ can represent the camera information without requirement of training an extra camera model on the test set. We also try to train a camera model on the training set, but it performs worse than computing $\overline{g}_C$.

In addition, since the tracklet ID of each image is provided, we can integrate all features of the tracklet T to get the tracklet feature t_I of I. Finally, the final feature of I is computed as follow:

$$\hat{f}_I = \beta f_I + (1-\beta) t_I \quad , \tag{3}$$

where β is the balance weight between single-frame and tracklet features. $\hat{f}_I$ is fed into a cluster to generate pseudo labels. Similar with previous works [4, 5], DBSCAN is chosen as the cluster in this paper. Then, we fine-tune baseline models trained in the Stage1. In the second stage (Stage2), all models are trained for only 3 epochs with a small learning rate. β is set to 0.0005 for the UDA training.

3.3. Post-Processing

Post-processing can significantly improve ReID performance in the inference stage. In this section, we will introduce several post-processing methods in this paper.

Augmentation Test. For each test image, we extract features of original image (CVF2) and cropped image (CFV2-C). Then, we flip these two images horizontally and additionally extract two features. We get four features totally and then average them to get the final ReID feature.

Re-Ranking. We adopt a widely used Re-ranking (RK) method [31] to update the final result. we set $k1 = 7$, $k2 = 2$, $\lambda = 0.6$ in this paper.

Weighted Tracklet Features. The tracklet IDs are provided for the testing set in Track2. A prior knowledge is that all frames of a tracklet belong to the same ID. In the inference stage, standard ReID task is an image-to-image

(I2I) problem. However, with the tracklet information, the task becomes an image-to-track (I2T) problem. For the I2T problem, the feature of tracklet is represented by features of all frames of the tracklet. He *et al*. [11] compared average features (AF) and weighted features (WF) of tracklets. In this paper, we calculated the weighted features for each tracklet.

Camera Verification. Zheng *et al*. [30] 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 ID from candidate images.

Inter-Camera Fusion. As we introduce in Eq 3.2, we also update a ReID feature with its inter-camera feature in the inference stage. The camera bias can be reduced. β is set to 0.18 for the post-processing.

Camera and Orientation Bias. Inspired by [30, 34], we train a camera model and a orientation model on CityFlow-V2 and VehicleX, respectively. Because VechcleX has orientation label available, we split the angle (0 - 360) into 36 bins, each bin is treated as an orientation ID. After training camera and orientation models, we can calculate camera and orientation similarities between an image pair. The fusion distance matrix can be expressed as following [34]:

$$D = D_r - \lambda_1 D_c - \lambda_2 D_o, \quad , \tag{4}$$

where D_r, D_c and D_o are ID distance matrix, orientation distance matrix and camera distance matrix, respectively. In this paper, we set $\lambda_1 = 0.1$ and $\lambda_2 = 0.05$.

4. Experiments

4.1. Implementation Details

CNN-Based Models in Stage1. All the images are resized to 384 $\times$ 384. As for data augmentation, we use random flipping, random padding and random erasing. In the training stage, we use soft margin triplet loss with the mini-batch of 8 identities and 12 images of each identity which leads to better convergence. SGD is adopted as the optimizer and the initial learning rate is set to $1e^{-2}$. Besides, we adopt the Warmup learning strategy and spend 10 epochs linearly increasing the learning rate from $1e^{-3}$ to $1e^{-2}$. The learning rate is decayed to $1e^{-3}$ and $1e^{-4}$ at 40th and 70th epoch, respectively. We totally train the model with 80 epochs. We adopt ResNet-IBN [8, 26], DenseNet-IBN [12, 26], ResNest [28], SeResNet-IBN [26] and ResNext-IBN [26] as backbones. All backbone are pre-trained on ImageNet [1].

Transformer-Based Models in Stage1. All the images are resized to 256 $\times$ 256 because of the limited GPU memory. The augmentation, batch size and optimizer are same with CNN-based models. However, we adopt cosine learning rate with only 40 epochs for TransReID because it converges faster than CNN models. The backbone is ViT/16-Base [3] pre-trained on ImageNet. In order to simplify our codes in this paper, the JPM and SIE modules are not used.

UDA Training in Stage2. In the stage2, the DBSCAN is adopted as the cluster. All models are fine-tuned for only three epochs. The learning rate is fixed to $4e^{-4}$. To increase diversity, each model is fine-tuned with two sets of pseudo labels generated by two different cluster parameters.

4.2. Validation Data

Since each team has only 20 submissions, it is necessary to use the validation set to evaluate methods offline. We split the training set of CityFlow-V2 into the training set and the validation set. For convenience, the validation set is denoted as Split-Test. Split-Test includes 18701 images of 88 vehicles.

4.3. The Ablation Study of Training Data

Datasets	CVF2	CVF2-C	VeX	VeX-S	mAP	R-1
Type1	✓				36.0	51.7
Type2	✓		✓		46.2	64.8
Type3	✓	✓	✓		49.1	65.9
Type4	✓	✓		✓	49.9	66.4

Table 1. The performance of models trained with different training sets on Split-Test. The backbone is ResNet50-IBN-a and all training images are resized to 256 $\times$ 256.

To explore the effectiveness of training data, we compare models trained with different training sets in Table 1. We train a ResNet50-IBN-a model with images resized to 256 $\times$ 256 as the baseline. When we only use CVF2 to train the model, it achieves 36.0% mAP on Split-Test. The synthetic dataset VeX improves the performance by a large margin, which achieves 46.2% mAP on Slpit-Test. In addition, the cropped data CVF2-C brings 2.9% mAP gains because of relatively loose cropping in CityFlow-V2. However, the 'real-world' data VeX-S generated by SPGAN does not surpass VeX by a lot. We infer that it may be because we did not adjust the parameters of SPGAN on the CityFlow-V2, so the quality of the generated data is not very high.

4.4. Comparison of Different Backbones

We comparison two CNN-based backbones (*i.e*. ResNet50-IBN-a and ResNet101-IBN-a) and a transformer-based backbone (*i.e*. TransReID) in Table 2. For a fair comparison, all training images are resized to 256 $\times$ 256.

When we only use CityFlow-V2 to train models, TransReID achieves better performance than ResNet50-IBN-a and ResNet101-IBN-a, which shows that the transformer-based models has a strong representation ability. However, when the synthetic dataset VehicleX is added into training, TransReID obtains the worst performance in these three

Datasets	Backbones	mAP	R-1
Type1	ResNet50-IBN-a	36.0	51.7
Type1	ResNet101-IBN-a	38.8	54.8
Type1	TransReID	42.1	59.8
Type2	ResNet50-IBN-a	46.2	64.8
Type2	ResNet101-IBN-a	47.6	68.2
Type2	TransReID	45.5	61.0

Table 2. The performance of different backbones is compared on Split-Test. All training images are resized to 256×256. Type1 means we use ony CityFlow-V2 to train models. Type2 means we use both CityFlow-V2 and VehicleX to train models.

models. The reason may be that TransReID is easier to overfit to the training data than CNN-based backbones.

4.5. The Ablation Study of Two-Stage Training

To evaluate the effectiveness of UDA training in the Stage2, we conduct experiments on the ResNet50-IBN-a baseline in the Table 3. DBSCAN clusters all test images into approximately 900 IDs. After UDA training, the performance of the model increases from 46.2% to 66.0% mAP. Due to the limited time, we did not verify the influence of inter-camera fusion and tracklet features too much. But both of them can significantly improve quality of pseudo labels, and then improve the performance of the model. Another experience we have observed is that fine-tuning the model too many epochs will harm the performance of the model.

Methods	mAP	R-1
Stage1	46.2	64.8
Stage2	66.0	76.8

Table 3. The ablation study of two-stage training on Split-Test. The backbone is ResNet50-IBN-a.

4.6. The Ablation Study of Post-Processing

We show the ablation study of post-processing in Table 4. The baseline is ResNet50-IBN-a trained on CityFlow-V2 and VehicleX. These post-processing methods improve the performance by almost 30% mAP totally on Split-Test, which shows the effectiveness of post-processing methods.

4.7. Ablation Study of the Solution

We present the ablation study of different parts in Table 5. Most of results have been present in aforementioned tables. In overview, training data, uda training, post-processing improve the performance by about 10%, 20%, 30% mAP on Split-Test, respectively. However, post-processing methods only improve 4.3% mAP and 5.0% Rank-1 accuracies on Split-Test after the UDA training. For

Methods	mAP	R-1
Baseline	46.2	64.8
+ Augmentation Test	47.0	65.0
+ Re-Ranking	58.8	66.1
+ Weighted Tracklet Features	66.8	66.9
+ Camera Verification	72.7	73.3
+ Inter-Camera Fusion	74.7	75.4
+ Camera and Orientation Bias	76.0	75.9

Table 4. The ablation study of post-processing methods on Split-Test. With all post-processing methods, ResNet50-IBN-a achieves 0.6055 mAP score on the leaderboard.

Methods	mAP	R-1
Baseline (Type1)	36.0	51.7
Baseline (Type3)	46.2	64.8
+ UDA Training	66.0	76.8
+ Post-processing	76.0	75.9
+ UDA Training & Post-processing	80.3	80.9

Table 5. The ablation study of all parts on Split-Test. With the UDA training and post-processing methods, ResNet50-IBN-a achieves 0.6665 mAP score on the leaderboard.

	Stage1		Stage2		Post	
Backbones	mAP	R-1	mAP	R-1	mAP	R-1
ResNet101-IBN-a	54.0	72.3	71.2	79.2	81.6	83.5
ResNet101-IBN-a†	54.9	72.7	72.5	80.2	82.2	83.5
ResNet101-IBN-a‡	54.3	71.5	71.4	79.0	82.6	84.6
ResNext101-IBN-a	50.6	70.2	74.0	82.5	82.5	84.0
ResNest101	51.2	70.8	72.4	80.8	82.5	84.1
SeResNet101-IBN-a	51.6	69.9	73.1	81.6	82.1	84.3
DenseNet169-IBN-a	51.2	69.2	69.9	79.6	82.7	85.9
TransReID	47.5	65.2	52.4	70.3	78.5	81.3
Ensemble (CNN)	-	-	-	-	84.3	85.9
Ensemble (All)	-	-	-	-	84.8	86.7

Table 6. Detailed results of single models and model ensemble on Split-Test. 'Post' means Post-Processing. † means the training data is Type4 (*i.e.* CFV2 + CFV2-C + VeX-S). ‡ means the augmentation is a little different. 'Ensemble (CNN)' means we ensemble 7 CNN models apart from TransReID. 'Ensemble (All)' means we ensemble all 8 models. ResNext101-IBN-a achieves 0.7058 mAP score on the leaderboard.

reference, 'Baseline (Type3) + Post-processing' achieves 0.6055 mAP score. The UDA training improve the score to 0.6665 on the leadeborad.

4.8. Model Ensemble

We ensemble several models to boost the performance on the leaderboard. ResNet101-IBN-a, DenseNet169-INB-a, ResNext101-IBN-a, SeResNet101-IBN-a, ResNest101 and TransReID are adopted as backbones. Unless otherwise specified, the training data is set to Type3 (*i.e.* CFV2 +

Rank	Team ID	Team Name	mAP Scores
1	**47**	**DMT (Ours)**	**0.7445**
2	9	NewGeneration	0.7151
3	7	CyberHu	0.6650
4	35	For Azeroth	0.6555
5	125	IDo	0.6373
6	44	KeepMoving	0.6364
7	122	MegVideo	0.6252
8	71	aiem2021	0.6216
9	61	CybercoreAI	0.6134
10	27	Janus Wars	0.6083

Table 7. Competition results of AICITY21 Track2.

CFV2-C + VeX) in the Stage1. With different training settings, we train 8 models in the Stage1. We then adopt the UDA training to fine-tine these models in the Stage2. Detailed results are present in Table 6. An interesting phenomena we observed is that TransReID can provide representation diversity different from CNN models. When we ensemble 7 CNN models, Ensemble (CNN) achieves 84.3% mAP and 85.9% rank-1 accuracy on Split-Test. However, when we continue to integrate TransReID, the performance increases to 84.8% mAP and 86.7% rank-1 accuracy even that TransReID achieves only 78.5% mAP and 81.3% rank-1 accuracy. Therefore, the diversity of TransReID plays an important role in the model ensemble.

It is note that we try two different cluster parameters for each model in the Stage2. Thus we train totally 16 models. In our experience, the performance of a single model ranges from 0.69 to 0.72 mAP scores. For instance, ResNext101-IBN-a achieves 0.7058 mAP score. We ensemble these 16 models, and achieves 0.7445 mAP score on the final leaderboard, yielding the first place in the Track2.

4.9. Competition Results

Our team (Team ID 47) achieves 0.7445 in the mAP score which achieves the first place in the 2021 NVIDIA AI City Challenge Track 2. As shown in the Table 7, it is the performance of top-10 teams. Our codes are available at https://github.com/michuanhaohao/AICITY2021_Track2_DMT.

5. Conclusion

In this paper, we conduct an empirical study of vehicle ReID on the 2021 AI City Challenge. We verify the UDA training is important in this challenge. In addition, transformer-based models are first time to be studied in the AI City Challenge. We believe transformer-based methods have great potential for ReID tasks. Finally, our solution yield the first place in the Track2 of the 2021 AI City Challenge.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5

[2] Weijian Deng, Liang Zheng, Qixiang Ye, Guoliang Kang, Yi Yang, and Jianbin Jiao. Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 994–1003, 2018. 1, 4

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 5

[4] Yixiao Ge, Dapeng Chen, and Hongsheng Li. Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification. *arXiv preprint arXiv:2001.01526*, 2020. 4

[5] Jianyang Gu, Hao Luo, Weihua Chen, Yiqi Jiang, Yuqi Zhang, Shuting He, Fan Wang, Hao Li, and Wei Jiang. 1st place solution to visda-2020: Bias elimination for domain adaptive pedestrian re-identification. *arXiv preprint arXiv:2012.13498*, 2020. 4

[6] Haiyun Guo, Kuan Zhu, Ming Tang, and Jinqiao Wang. Two-level attention network with multi-grain ranking loss for vehicle re-identification. *IEEE Transactions on Image Processing*, 28(9):4328–4338, 2019. 2

[7] Bing He, Jia Li, Yifan Zhao, and Yonghong Tian. Part-regularized near-duplicate vehicle re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3997–4005, 2019. 2

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[9] Shuting He, Hao Luo, Weihua Chen, Miao Zhang, Yuqi Zhang, Fan Wang, Hao Li, and Wei Jiang. Multi-domain learning and identity mining for vehicle re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 582–583, 2020. 2, 3

[10] Shuting He, Hao Luo, Pichao Wang, Fan Wang, Hao Li, and Wei Jiang. Transreid: Transformer-based object re-identification. *arXiv preprint arXiv:2102.04378*, 2021. 2, 3

[11] Zhiqun He, Yu Lei, Shuai Bai, and Wei Wu. Multi-camera vehicle tracking with powerful visual features and spatial-temporal cue. In *Proc. CVPR Workshops*, pages 203–212, 2019. 5

[12] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 5

[13] Pirazh Khorramshahi, Amit Kumar, Neehar Peri, Sai Saketh Rambhatla, Jun-Cheng Chen, and Rama Chellappa. A dual-path model with adaptive attention for vehicle re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6132–6141, 2019. 2

[14] Pirazh Khorramshahi, Neehar Peri, Amit Kumar, Anshul Shah, and Rama Chellappa. Attention driven vehicle re-identification and unsupervised anomaly detection for traffic understanding. In *Proc. CVPR Workshops*, pages 239–246, 2019. 2

[15] Hongye Liu, Yonghong Tian, Yaowei Wang, Lu Pang, and Tiejun Huang. Deep relative distance learning: Tell the difference between similar vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2167–2175, 2016. 2

[16] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of tricks and a strong baseline for deep person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2

[17] Hao Luo, Wei Jiang, Youzhi Gu, Fuxu Liu, Xingyu Liao, Shenqi Lai, and Jianyang Gu. A strong baseline and batch normalization neck for deep person re-identification. *IEEE Transactions on Multimedia*, 2019. 2

[18] Milind Naphade, Zheng Tang, Ming-Ching Chang, David C. Anastasiu, Anuj Sharma, Rama Chellappa, Shuo Wang, Pranamesh Chakraborty, Tingting Huang, Jenq-Neng Hwang, and Siwei Lyu. The 2019 AI City Challenge. In *Proc. CVPR Workshops*, pages 452–460, 2019. 3

[19] Milind Naphade, Shuo Wang, David C Anastasiu, Zheng Tang, Ming-Ching Chang, Xiaodong Yang, Liang Zheng, Anuj Sharma, Rama Chellappa, and Pranamesh Chakraborty. The 4th ai city challenge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 626–627, 2020. 2

[20] Jingjing Qian, Wei Jiang, Hao Luo, and Hongyan Yu. Stripe-based and attribute-aware network: A two-branch deep model for vehicle re-identification. *arXiv preprint arXiv:1910.05549*, 2019. 2

[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 *Proceedings of the IEEE International Conference on Computer Vision*, pages 1900–1909, 2017. 2

[22] 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 *Proc. CVPR*, pages 8797–8806, 2019. 1, 3

[23] Shangzhi Teng, Xiaobin Liu, Shiliang Zhang, and Qingming Huang. Scan: Spatial and channel attention network for vehicle re-identification. In *Pacific Rim Conference on Multimedia*, pages 350–361. Springer, 2018. 2

[24] Peng Wang, Bingliang Jiao, Lu Yang, Yifei Yang, Shizhou Zhang, Wei Wei, and Yanning Zhang. Vehicle re-identification in aerial imagery: Dataset and approach. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 460–469, 2019. 2

[25] 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 *Proceedings of the IEEE International Conference on Computer Vision*, pages 379–387, 2017. 2

[26] Jianping Shi Xingang Pan, Ping Luo and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *ECCV*, 2018. 5

[27] Yue Yao, Liang Zheng, Xiaodong Yang, Milind Naphade, and Tom Gedeon. Simulating content consistent vehicle datasets with attribute descent. arXiv:1912.08855, 2019. 1, 4

[28] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Zhi Zhang, Haibin Lin, Yue Sun, Tong He, Jonas Mueller, R Manmatha, et al. Resnest: Split-attention networks. *arXiv preprint arXiv:2004.08955*, 2020. 5

[29] Zhedong Zheng, Minyue Jiang, Zhigang Wang, Jian Wang, Zechen Bai, Xuanmeng Zhang, Xin Yu, Xiao Tan, Yi Yang, Shilei Wen, et al. Going beyond real data: A robust visual representation for vehicle re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 598–599, 2020. 2

[30] Zhedong Zheng, Tao Ruan, Yunchao Wei, and Yezhou Yang. Vehiclenet: Learning robust feature representation for vehicle re-identification. In *CVPR Workshops*, volume 2, page 3, 2019. 2, 5

[31] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1318–1327, 2017. 2, 4

[32] Yi Zhou and Ling Shao. Aware attentive multi-view inference for vehicle re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6489–6498, 2018. 2

[33] Jianqing Zhu, Huanqiang Zeng, Jingchang Huang, Shengcai Liao, Zhen Lei, Canhui Cai, and Lixin Zheng. Vehicle re-identification using quadruple directional deep learning features. *IEEE Transactions on Intelligent Transportation Systems*, 2019. 2

[34] Xiangyu Zhu, Zhenbo Luo, Pei Fu, and Xiang Ji. Voc-reid: Vehicle re-identification based on vehicle-orientation-camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 602–603, 2020. 2, 3, 5