A Strong Baseline for Vehicle Re-Identification

Su V. Huynh, Nam H. Nguyen, Ngoc T. Nguyen, Vinh TQ. Nguyen, Chau Huynh, Chuong Nguyen
Cybercore AI
{su.huynh, nam.nguyen, ngoc.nguyen, vinh.nguyen, chau.huynh, chuong.nguyen}@cybercore.co.jp

Abstract

*Vehicle Re-Identification (Re-ID) aims to identify the same vehicle across different cameras, hence plays an important role in modern traffic management systems. The technical challenges require the algorithms must be robust in different views, resolution, occlusion and illumination conditions. In this paper, we first analyze the main factors hindering the Vehicle Re-ID performance. We then present our solutions, specifically targeting the dataset Track 2 of the 5th AI City Challenge, including (1) reducing the domain gap between real and synthetic data, (2) network modification by stacking multi heads with attention mechanism, (3) adaptive loss weight adjustment. Our method achieves **61.34**% mAP on the private CityFlow testset without using external dataset or pseudo labeling, and outperforms all previous works at **87.1**% mAP on the Veri benchmark. The code is available at* `https://github.com/cybercore-co-ltd/track2_aicity_2021`.

1. Introduction

Vehicle Re-ID aims to re-target vehicle images across non-overlapping camera views given a query image. It has many practical applications, such as for analyzing and managing the traffic flows in Intelligent Transport System.

Despite many progresses have been made in the recent years thanks to deep learning, vehicle Re-ID is still facing many challenges, such as severe variations from different view points, partial occlusion, image blurry or illumination changes. The state-of-the-art methods [25, 30, 4] typically use a deep neural network to extract the vehicle visual representation. Some methods proposed to enhance the feature representation by using multi-head architecture to extract multi-scale information, such as Zheng *et al.* [25]. However, they only use simple pooling operators to extract feature vectors, which then be averaged in inference stage. Hence, the feature lacks the vehicle detailed characteristics, which is important to distinguish objects with similar appearance.

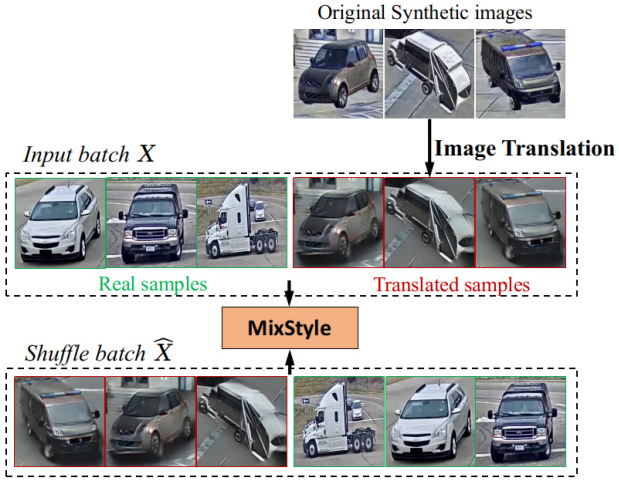

Figure 1: Domain Generalization with MixStyle.

In contrast, in Face ID application, Kim *et al.* [6] propose to retain the detailed characteristics into several latent groups, which improve the Re-ID confidence. This motivates us to develop a mechanism to integrate the feature vectors into several common groups, which help filtering out the candidates during retrieval. To improve the model generalization, a typical practice is to train the model on large datasets, such as the Veri dataset [9] and PKU-VehicleID dataset [7]. Another approach to obtain large dataset without costly human labeling is to utilize synthetic data generated from 3D simulation environment, by which we can have fully control of the vehicle's appearance [21]. In the Track 2 of the 5th AI City Challenge, two datasets are provided, namely the real-world and synthetic data, as illustrated in Figure 1. However, as seen in Figure 1, there is always a domain gap between two data sources, which leads to feature distribution shifting. To tackle this problem, Zheng *et al.* [25] adopt the image translation technique (UNIT [8]) to transform the synthetic data closer to the realistic one. However, the translated images are still in poor quality, and the domain gap is still significant.

In addition, designing effective loss functions to train the network is also very important. A majority of previous

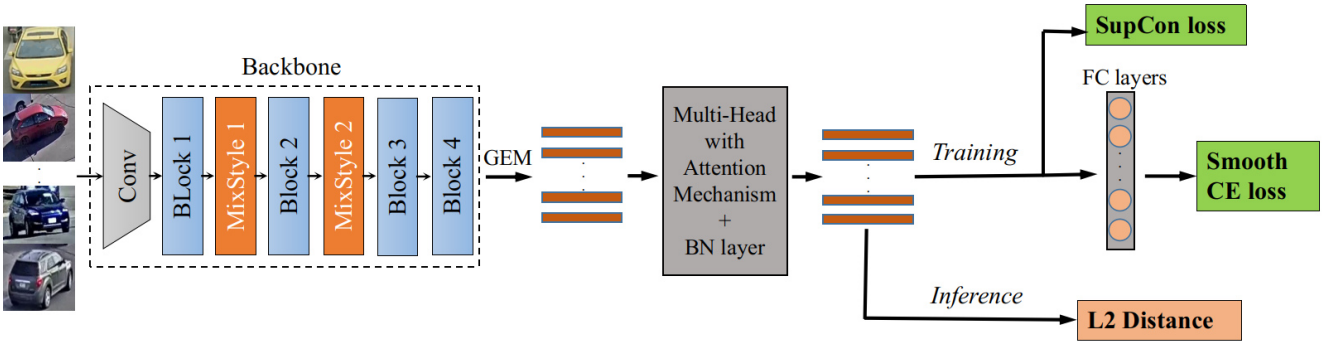

Figure 2: The training pipeline. GEM: Generalized Mean Pooling, BN: Batch Normalization, FC: Fully Connected, CE: Cross Entropy, SupCon: Supervised Contrastive.

works [26, 25, 30, 4] typically use a combination of Triplet Loss and Cross Entropy Loss. The loss function 's objective is intuitive: pulling samples with the same ID together, while pushing those with different IDs far apart. However, the Triplet Loss function only uses one positive and one negative pair per anchor, and the hard negative mining process must be tuned carefully. Moreover, the ratio between Triplet Loss and Cross Entropy Loss is heuristically set to 1:1. However, our experiments show that this ratio setting also has a strong impact to the performance, but surprisingly, to the best of our knowledge, it is often overlooked in the previous works.

From the aforementioned analysis, we present our solutions to adress the problem. Our main contributions are:

(1) We adopt MixStyle Transfer [29] as a regularization method to reduce the gap between the real and synthetic data.

(2) Multi-head with attention mechanism are attached to the backbone to help the model learn more detailed features. The features are then automatically grouped into subfeatures, each help narrows down the search space of the target identity.

(3) We replace the commonly used Triplet Loss with the Supervised Contrastive Loss [5] which help the network learning more effectively. Additionally, a novel adaptive loss weight between the Supervised Contrastive Loss and the Cross Entropy Loss is provided to improve the performance dramatically.

2. Related work

In order to enrich visual representation for deep learning based models, a large scale dataset is necessary. Liu *et al.* [9] propose the VeRi dataset, which contains a large number of vehicles captured by non overlapping cameras with different perspectives, scales and illuminations in real world urban traffic. In addition, annotating dataset is very costly and time consuming. To solve this problem, many efforts have been made to improve the data generation techniques. For instance, Yao *et al.* [21] introduce a large-scale synthetic dataset simulated by a flexible 3D graphic engine with editable attributes such as vehicle orientation, light direction and camera height. Recently, the generative adversarial network (GAN) can be used to generate new data by transferring vehicle style [1, 18] or changing the vehicle attributes [27]. Moreover, Zhou *et al.* propose the MixStyle method [29], which attempts to create a domain-generalized model by mixing the feature statistics to simulate new styles. However, simply adding synthetic data to train the model often yeilds inferior results, due to the domain gap and feature bias between the synthetic and the real world data.

Other methods focus on developing more effective loss functions to improve the network training efficiency. For instance, the Large Margin Cosine Loss [17] aims to maximize the inter-class variance and minimize intra-class variance. The Triplet Loss [19] aims to learn visual representation by optimizing the distances between a set of three hard samples. Sun *et al.* [12] propose the Circle Loss, which adaptively adjusts weights for each similarity score. In addition to loss functions, sampling strategy also plays an important role in re-id training. The Hierarchical Triplet Loss [2] uses a predefined hierarchical tree to formulate informative training samples, which help to overcome the limitation of random sampling when training triplet loss. The semi-hard triplet mining [11] focuses on negative examples which have close distances to the anchor positive distances. However, sampling strategy is generally heuristic, depending on the loss function, and hard to tune.

Additionally, post-processing is also important to reduce the false-positive prediction. For example, re-ranking can improve the accuracy of the ranking list. Re-ranking approaches are widely used in person re-id [22], [10], which typically rely on the consistency and nearest-neighbor relationship of gallery images based on initial re-id ranking. Recently, Zhong *et al.* [28] propose the k-reciprocal encod-

ing method , which considers the original distance and the Jaccard distance between two images. In this work, we also perform an ablation study to find the best practice in applying post-processing steps to the vehicle Re-ID problem.

3. Proposed Method

In section 3.1, we introduce the algorithm to bridge the gap between synthetic data and real data. Then we show our baseline architecture which applies multi-head with attention mechanism in section 3.2. In section 3.3, the alternative Contrastive Loss and Adaptive Loss Weight are introduced. Finally, we present some bag of post-processing tricks in section 3.4.

3.1. Domain Generalization

To train a model that generalizes to unseen domains, we adopt the MixStyle method [29], which aims to simulate new styles by mixing the statistical features of two samples from different domains. Given the input batch X (i.e., real and synthetic samples in a same batch training) and a shuffle of X, named $\hat{X}$, MixStyle computes the mixed feature's statistics by

$$\mu_m = \lambda\mu(X) + (1-\lambda)\mu(\hat{X}) \quad (1)$$

$$\sigma_m = \lambda\sigma(X) + (1-\lambda)\sigma(\hat{X}) \quad (2)$$

where λ is the weights sampled from $Beta$ distribution, $\lambda \sim Beta(\alpha, \alpha)$. Following [29], we set $\alpha = 0.1$ throughout all our experiments. Rely on the mixed feature statistic, style-normalized X is computed as

$$\mathrm{MixStyle(X)} = \sigma_\mathrm{m}\frac{\mathrm{X}-\mu(\mathrm{X})}{\sigma(\mathrm{X})} + \mu_\mathrm{m}. \quad (3)$$

By leveraging the feature-level statistics, MixStyle implicitly regularizes the network. This makes the model become more robust to the domain difference and enforce the network to learn the object semantic features.

3.2. Network Architecture

We adopt the method proposed in [30] as the baseline, and augment it with our proposed network modification. The network architecture, the training and inference pipeline are illustrated in Figure.2.

Backbone. We use Instance Batch Normalization (IBN) network family [20] as the backbone due to its advantages. Firstly, by utilizing the instance normalization, the feature extractor can learn robust encoded representations that invariant to appearance differences. Secondly, it can improve the performance of other advanced deep neural network architecture such as ResNet, ResNeXt, and SENet. Moreover, we attempt to append MixStyle layers into the network to improve the domain generalization. Specifically, as described in [29], convolution layers in early stages encode the style information, while later stages tend to capture the semantic content. Therefore, we add the MixStyle module after the Block 1 and 2 in the ResNeXt_ibn_a 101 backbone [20], as shown in Figure 2.

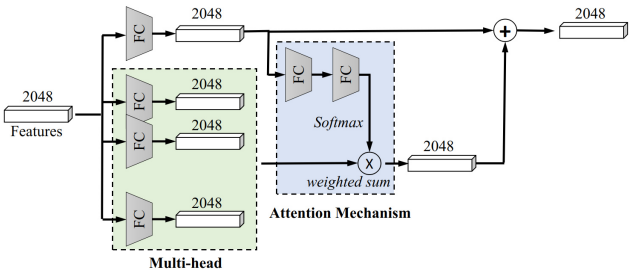

Figure 3: Multi-head with attention mechanism.

Multi-head with Attention Mechanism. Distinguishing thousands of vehicles with multiple views is challenging. Instead of using only one head, using multi-head encourages the re-id model to learn more diverse features from different vehicle characteristics. Thus, we adopted the multiples heads architecture [6] to further enhance the quality of the visual representation for vehicle re-id. Figure 3 shows the architecture of the multiple head with attention mechanism. In particular, the 2048-dim feature obtained from the backbone is fed into multiple parallel fully connected (FC) layers. Following [6], each FC layer is considered as one head and expected to learn distinct features which take into account different vehicle characteristics. Additionally, the attention mechanism determines which head's features are more important to the final encoding feature.

3.3. Loss Function

3.3.1 ID and Metric Losses

For training re-id model, a common approach is to use a combination of *ID Loss* and *Metric Loss*. In particular, Cross Entropy (CE) is used for ID Loss to classify samples in different classes, while Metric Loss is often the Contrastive Loss, such as Triplet Loss [19] or Circle Loss [13], to optimize the feature distance between each class.

ID Loss. In this work, we use the CE Loss for the ID Loss. Given an input image, the ID embedding vector is extracted from the fully connected layer attached to the multi-head module with the dimension equal to the number of vehicles N. Let y is the ground truth ID label and p_i is the ID prediction logits of class i, we use the Label Smoothing technique [15] to prevent the model from over-fitting and improve robustness, the Label Smoothing CE Loss is defined as:

$$\mathcal{L}(ID) = \sum_{i=1}^{N} -q_i log(p_i) \begin{cases} q_i = 0, y \neq i \\ q_i = 1, y = i \end{cases} \quad (4)$$

$$q_i = \begin{cases} 1 - \frac{N-1}{N}\varepsilon & \text{if } i = y \\ \varepsilon / N & \text{otherwise,} \end{cases} \quad (5)$$

where, ε is a soft-margin to reduce the model over-confidence and is set to 0.1 in our experiments.

Metric Loss. To improve the model performance on hard samples, we adopt the Supervised Contrastive Loss (SupCon) [5]. Specifically, the SupCon can be seen as a generalized case of the Triplet and N-pair loss. Instead of using only one positive and one negative pair for each anchor, the SupCon considers many positive and negative pairs. Applying SupCon to the ReID problem provides several benefits (1) the gradient of SupCon loss function encourages learning from hard positives and hard negatives; and (2) it is less sensitive to hyper-parameters. The SupCon is computed as:

$$\mathcal{L} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} log \frac{exp(z_i z_p / \tau)}{\sum_{a \in A(i)} exp(z_i z_a / \tau)}, \quad (6)$$

where $P(i) \equiv \{p \in A(i) : \tilde{\boldsymbol{y}}_p = \tilde{\boldsymbol{y}}_i\}$ is the set of indices of all positives in the multi viewed batch distinct from i, $|P(i)|$ is its cardinality, $\tau \in \mathcal{R}^+$ is a scalar temperature parameter, z_i is an anchor feature, z_p is a positive feature and z_a is a negative feature.

3.3.2 Constructing Adaptive Loss Weight

Problems in Training. Training the ReID model requires optimizing a combination of ID Loss and Metric Loss. Conventionally, the loss weights are set equally, i.e. 1:1 ratio. However, in practice, ID Loss is relatively much larger than Metric Loss, which causes the imbalance and affects the training performance. Table 1 shows the sensitiveness of performance towards loss weight. Unfortunately, manually tuning the loss weight is sub-optimal and time consuming. Hence, motivated by the Adaptive Loss Weight Adjustment[23], we propose the Momentum Adaptive Loss Weight (MALW) to increase training stability by automatically updating loss weights according to the statistical characteristics of loss values.

Loss weight	1:1	1:2	0.5:0.5	MALW
mAP(%)	73.3	75.2	76.8	**78.4**

Table 1: The performance of Baseline from [30] under different loss weights (Cross Entropy Loss weight:Triplet Loss weight) and MALW.

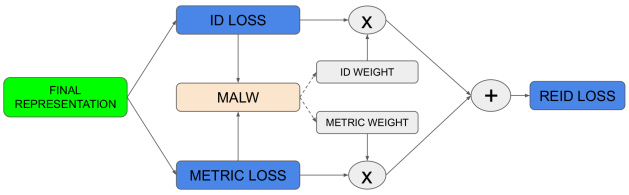

Figure 4: Momentum Adaptive Loss Weight (MALW).

Algorithm 1 Momentum Adaptive Loss Weight

Input: - ID Loss weight λ_{ID}
- Metric Loss weight λ_{Metric}
- Update iteration count k
- Momentum factor α

Begin:
Initialize loss weight λ_{ID}, λ_{Metric} to $1:1$
Build two empty sets S_{ID}, S_{Metric} for recording losses

for $i = 0$ to max iter **do**
  Obtain initial loss L_{ID} and L_{Metric}
  Set $L_{ID} = \lambda_{ID} L_{ID}$, $L_{Metric} = \lambda_{Metric} L_{Metric}$
  Add L_{ID} to S_{ID} and L_{Metric} to S_{Metric}
  if $i\%k == 0$ **then**
    $ID_{std} = std(S_{ID})$, $Metric_{std} = std(S_{Metric})$
    Empty sets S_{ID}, S_{Metric}
    if $ID_{std} > Metric_{std}$ **then**
      $new\_\lambda_{ID} = 1 - (ID_{std} - Metric_{std})/ID_{std}$
      Update weight: $\lambda_{ID} = \alpha * \lambda_{ID} + new\_\lambda_{ID}$
    end if
  end if
end for

Output: $(\lambda_{ID}, \lambda_{Metric})$

Momentum Adaptive Loss Weight. Algorithm 1 and Figure 4 describe how the MALW updates the weights during training progress. Let λ_{ID} and λ_{Metric} be the loss weights for ID Loss and Metric Loss, respectively. Initially, the ratio between λ_{ID} and λ_{Metric} is set to 1:1. After K iterations training, the ID loss weight λ_{ID} is updated based on the standard deviation of the recorded ID Loss L_{ID} and Metric Loss L_{Metric} with a momentum factor. The MALW method improves our model performance by balancing the training losses without adding any computation cost to the inference step, as seen in Table 1.

3.4. Post-Processing

Re-rank using k-reciprocal encoding. We improve performance of the re-id model using a re-ranking method described in [28]. This approach refines the initial ranking list using the information of original distance and Jaccard distance between two vehicle images.

Fused distance. To reduce the influence of vehicle orientations and camera viewpoints, we adopt the fusion technique proposed in [30]. Specifically, the vehicle ID, orientation and camera distance matrices are fused to get the cost fusion matrix, which is then used to find the optimal results for query images, as:

$$D(x_i, x_j) = D_v(x_i, x_j) - \lambda_1 D_o(x_i, x_j) - \lambda_2 D_c(x_i, x_j), \tag{7}$$

where $D_v(x_i, x_j)$, $D_o(x_i, x_j)$, $D_c(x_i, x_j)$ are ID distance, orientation distance and camera distance between two vehicle images (x_i, x_j) , respectively.

Tracklet-Level Re-Ranking. Additionally, we apply another re-ranking method using the tracklet information of vehicles which is included in the CityFlow dataset. To be specific, a vehicle 's tracklet is created from the detection and tracking results in one camera. Replacing features of each image in a tracklet with averaging the features of a subset of consecutive frames can help us enhance the visual representation of the same vehicle [30].

Ensemble. We combined all 3 models using different backbones, including ResNet50_ibn_a [20], ResNeXt101_ibn_a [20], ResNet152 [3], by taking the averaged distance of each query image to gallery images. As shown in Table 4, our ensemble model significantly increases 2.8% mAP on the CityFlow test set.

4. Experimental Results

4.1. Data Analysis

The dataset of Track 2 challenge is the new version of CityFlow called CityFlowV2-ReID. There are 440 IDs retrieved from 52,717 images for training and other 440 identities come from 31.238 images in the test set. Following the restriction of using external data, our team tackles the problem of data limitation by leveraging the synthetic dataset called VehicleX [21]. There are totally 1362 unique identities and 192150 synthetic images in VehicleX dataset, which can be used for model training or transfer learning. We split the real training data into Split-Train and Split-Test to validate offline. In particular, Split-Train contains 44375 images of 360 IDs, while Split-Test includes 8342 images of 80 IDs.

4.2. Training Strategy

We resize the images to (320x320) and apply several data augmentation methods, such as color jitters, random flip, brightness and contrast adjustment, random erase and random cropping. We use ADAM [11] optimizer with the cosine annealing scheduler and set the learning rate to 3.5e-4. The batch size is set to 128, which compose of 16 identities, where each identity contains 8 images. For vehicle re-id model, we adopt three strong backbones: ResNet50_ibn_a [20], ResNeXt101_ibn_a [20] and ResNet152 [3] as our feature extractor. In addition, only ResNeXt101_ibn_a is used to train both Camera Re-ID and Orientation Re-ID models. All models are pretrained on ImageNet. For each single model, we first frozen the backbone and train multi heads for 1 epoch. Then we train the whole architecture for extra 11 epochs.

4.3. Ablation Study

In this section, we use Split-Train and Split-Test for ablation study. The results are summarized in Table 2, 3, and 4.

Baseline: We use the standard backbone ResNeXt101_ibn_a [20] along with CE Loss and Triplet Loss as a baseline for our vehicle re-id model, after training the baseline using the CityFlow dataset, we achieved 75.5% mAP and 22.1% mAP on Split-Test and CityFlow dataset.

Synthetic Data with MixStyle: To evaluate the effectiveness of using synthetic data, we train the baseline using the combination of real and synthetic data, result in the increment of mAP to 80.2% and 32.5% on Split-Test and CityFlow, respectively. This indicates that proper usage of synthetic data to train the network is helpful. Moreover, by using translated synthetic data instead of the original one, the mAP increases to reach 81.5% and 35.3%. We further alleviate the domain gap between these two data sources by applying MixStyle, gaining 2.3% and 2.4% more mAP score on these datasets, as shown in Table 2. This result opens up the possibility of using synthetic data for training deep re-id networks to reduce the cost of collecting real-word data and human annotation.

Multi-Head with Attention Mechanism: After getting the data strategy for training, we enhance the network capability by applying Multi-head with Attention Mechanism and achieve 84.5% and 41.9% mAP on Split-Test and CityFlow, respectively, as shown in Table 3. This demonstrates that the visual representation features obtained from multi-head are more robust compared to using only one single head.

Losses: The combination of CE Loss and Triplet Loss is widely used in re-id tasks. Here, replacing the Triplet Loss by the Supervised Contrastive Loss [5] results in the increment of 1.2% mAP, from 84.5% to 85.7% on Split-Test set. Moreover, the MALW is applied to balance the loss functions, which solves the slow convergence problem and eliminate the need of loss weight setting. We set $K = 500$ and

Data	Real-split		CityFlow	
	mAP(%)	Rank 1(%)	mAP(%)	Rank 1(%)
Real	75.5	79.7	22.1	31.6
Real + Syn	80.2	85.2	32.5	51.8
Real + Syn (translated)	81.5	86.7	35.3	54.3
Real + Syn (translated) + Mixstyle	83.8	88.7	37.7	58.2

Table 2: Different datasets on Real-split and CityFlow.

Method	Real-split		CityFlow	
	mAP(%)	Rank 1(%)	mAP(%)	Rank 1(%)
Baseline + Multiple Head	84.5	89.0	41.9	65.6
Baseline + Multiple Head + SupCon	85.7	90.9	-	-
Baseline + Multiple Head + SupCon + MALW	88.1	92.5	49.5	58.2

Table 3: Different training methods on Real-split and CityFlow.

Method	Performance			
Re-rank	✓	✓	✓	✓
Orientation & Camera ID		✓	✓	✓
Track-rank ReID			✓	✓
Ensemble				✓
mAP(%)	49.5	53.7	58.5	61.3
Rank 1(%)	58.2	64.7	70.2	72.2

Table 4: Different pos-process techniques on CityFlow.

Data	Veri dataset	
	mAP(%)	Rank 1 (%)
Strong Baseline [14]	67.6	90.2
DMML [24]	70.1	90.2
PAMTRI(ALL) [16]	71.8	92.8
VOC ReID [30]	82.8	**97.6**
Our	**87.1**	97.0

Table 5: Comparison with the state-of-the art methods on the VeRi776 dataset.

$\alpha = 0.9$ and this helps improve our mAP to 88.1% and 49.5% on Split-Test and CityFlow, respectively. The results are summarized in Table 3.

Post-Processing: Table 4 illustrates the results of applying different post-processing methods. Firstly, the re-ranking algorithm [28] is widely used and demonstrated its improvement, therefore, by default we apply it to all models. Secondly, the fused distance approach using the vehicle ID, Orientation and Camera distances [30] increases mAP from from 49.5% to 53.7%, which indicates that the Orientation and Camera information is useful for the ReID performance. Thirdly, by applying track-ranking algorithm, we further gain 4.8% mAP. Finally, after ensembling our three best single models, we achieve 61.34% mAP on the CityFlow test set without using any external data or pseudo tricks.

4.4. Performance on VeRi776

To further demonstrate the generalization across datasets, we also test our proposed method on the Veri benchmark dataset. For a fair comparison, we only use single model, including backbone ResNeXt101_ibn_a, multi-head, CE and SupCon losses and MALW without applying pos-processing technique and synthetic data. We achieve the state-of-the-art performance with a large margin compared to previous works, as shown in Table 5.

4.5. Visualization of results

We visualize the query images and ranking lists obtained by the baseline model and our final model. As shown in Figure 5, the baseline model fails to retrieve an accurate ranking list, since identifying vehicle objects from these query images is truly challenging. For example, the vehicle in the first row is occluded. Vehicles in the second and third row have similar appearance to other vehicles in the dataset, while the samples in the last row has very low resolution. On the contrary, our model surpasses the baseline and can retrieve a high quality ranking list, as shown Figure 6.

5. Conclusion

In this paper, we proposed a strong baseline for the vehicle re-identification problem. By making improvements on utilizing the usage of real and synthetic data, employing the multi-head with attention mechanism and optimizing a combination of training losses, we achieve 61.34% mAP on the CityFlow dataset. In the VeRi dataset, we achieve 87.1% mAP, outperform the previous works with a large margin. Our method is simple, and focuses on improving the training techniques more efficiently. Hence, it can be

Figure 5: Result on the baseline model. Each row presents the query images and retrieved top 6 gallery images. Green and red boxes denote true positive and false positive sample, respectively.

Figure 6: Result on the final model. Each row presents the query images and retrieved top 6 gallery images. Green and red boxes denote true positive and false positive sample, respectively.

generally applied to a variety of Re-ID problems. We also released the code to facilitate the reproduction, hoping that it can serve a new baseline for further research.

References

[1] 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 (CVPR)*, June 2018.

[2] Weifeng Ge, Weilin Huang, Dengke Dong, and Matthew R. Scott. Deep metric learning with hierarchical triplet loss. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 272–288, Cham, 2018. Springer International Publishing.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[4] 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 *Proc. CVPR Workshops*, 2020.

[5] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 18661–18673. Curran Associates, Inc., 2020.

[6] Yonghyun Kim, Wonpyo Park, Myung-Cheol Roh, and Jongju Shin. Groupface: Learning latent groups and constructing group-based representations for face recognition. pages 5620–5629, 06 2020.

[7] H. Liu, Y. Tian, Y. Wang, L. Pang, and T. Huang. Deep relative distance learning: Tell the difference between similar vehicles. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2167–2175, 2016.

[8] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. *CoRR*, abs/1703.00848, 2017.

[9] X. Liu, W. Liu, H. Ma, and H. Fu. Large-scale vehicle re-identification in urban surveillance videos. In *2016 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2016.

[10] M. S. Sarfraz, A. Schumann, A. Eberle, and R. Stiefelhagen. A pose-sensitive embedding for person re-identification with expanded cross neighborhood re-ranking. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 420–429, 2018.

[11] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, 2015.

[12] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[13] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[14] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. Beyond part models: Person retrieval with refined part pooling. *CoRR*, abs/1711.09349, 2017.

[15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, Los Alamitos, CA, USA, jun 2016. IEEE Computer Society.

[16] Zheng Tang, Milind Naphade, Ming-Yu Liu, Xiaodong Yang, Stan Birchfield, Shuo Wang, Ratnesh Kumar, David C. Anastasiu, and Jenq-Neng Hwang. Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. *CoRR*, abs/1903.09254, 2019.

[17] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[18] L. Wei, S. Zhang, W. Gao, and Q. Tian. Person transfer GAN to bridge domain gap for person re-identification. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 79–88, 2018.

[19] Kilian Q Weinberger, John Blitzer, and Lawrence Saul. Distance metric learning for large margin nearest neighbor classification. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems*, volume 18. MIT Press, 2006.

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

[21] Yue Yao, Liang Zheng, Xiaodong Yang, Milind Naphade, and Tom Gedeon. Simulating content consistent vehicle datasets with attribute descent. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part VI*, volume 12351 of *Lecture Notes in Computer Science*, pages 775–791. Springer, 2020.

[22] M. Ye, C. Liang, Y. Yu, Z. Wang, Q. Leng, C. Xiao, J. Chen, and R. Hu. Person reidentification via ranking aggregation of similarity pulling and dissimilarity pushing. *IEEE Transactions on Multimedia*, 18(12):2553–2566, 2016.

[23] Wenxin Yu, Bin Hu, Yucheng Hu, Tianxiang Lan, Yuanfan You, and Dong Yin. Revisiting the loss weight adjustment in object detection. *ArXiv*, abs/2103.09488, 2021.

[24] Liang Zheng, Hengheng Zhang, Shaoyan Sun, Manmohan Chandraker, and Qi Tian. Person re-identification in the wild. *CoRR*, abs/1604.02531, 2016.

[25] 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.

[26] Zhedong Zheng, Tao Ruan, Yunchao Wei, Yi Yang, and Tao Mei. Vehiclenet: Learning robust visual representation for vehicle re-identification. *IEEE Transactions on Multimedia (TMM)*, 2020.

[27] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint discriminative and generative learning for person re-identification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[28] Z. Zhong, L. Zheng, D. Cao, and S. Li. Re-ranking person re-identification with k-reciprocal encoding. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3652–3661, 2017.

[29] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *International Conference on Learning Representations*, 2021.

[30] Xiangyu Zhu, Zhenbo Luo, Pei Fu, and Xiang Ji. VOC-ReID: Vehicle re-identification based on vehicle-orientation-camera. In *Proc. CVPR Workshops*, 2020.