Attribute-guided Feature Extraction and Augmentation Robust Learning for Vehicle Re-identification

Chaoran Zhuge[1] , Yujie Peng[2] , Yadong Li[2] , Jiangbo Ai[3], Junru Chen[4]
University of Science and Technology of China[1]
School of Computer Science and Engineering, Beihang university[2]
University of Electronic Science and Technology of China[3]
Xidian University[4]

Abstract

Vehicle re-identification is one of the core technologies of intelligent transportation systems and smart cities, but large intra-class diversity and inter-class similarity poses great challenges for existing method. In this paper, we propose a multi-guided learning approach which utilizing the information of attributes and meanwhile introducing two novel random augments to improve the robustness during training. What's more, we propose an attribute constraint method and group re-ranking strategy to refine matching results. Our method achieves mAP of 66.83% and rank-1 accuracy 76.05% in the CVPR 2020 AI City Challenge.

1. Introduction

In recent years, the development of technology in the field of computer vision and the breakthrough of technology in the field of Internet of Things promote the realization of smart city concept. As important objects in smart city applications, vehicles have attracted extensive attention, a lot of researches about vehicles has been carried out, such as vehicle detection, vehicle tracking, fine-grained vehicle type recognition, etc.

As a frontier and important research topic, vehicle re-identification (ReID)[7] also caused more and more attention in research area, the purpose of vehicle ReID is to identify the same vehicle through multiple non-overlapping cameras. A vehicle ReID system can quickly get the location and time of the target vehicle in the city. Vehicle ReID technology is crucial to the future development of the Internet of things, as well as the construction of intelligent transportation system and smart city[23].

There are two main challenges of vehicle ReID, intra-class difference and inter-class similarity. Intra-class difference is mainly caused by viewpoint variation, what's more, background clutters, resolution and illumination also have great influence. Inter-class similarity is in reflected images of different vehicle may look very similar. Vehicles produced by the same or different manufacturers can have similar colors and shapes.

In this paper, our proposed method is focusing on extracting robust features for vehicle ReID task, Finally, ensemble and re-ranking is also used to refine the results.

In summary, our contributions are:

- we proposed a method utilizing attributes information for vehicle ReID.
- we introduce random shrink and background substitution augments to improve the robustness of model when the quality of images is largely different.
- we introduce an attribute constraint method and group re-ranking strategy to get more accurate matching results.
- Our vehicle ReID method achieves mAP of 66.83% and rank-1 accuracy 76.05% in the CVPR 2020 AI City Challenge.

2. Related Work

Similar to face recognition and pedestrian ReID, vehicle ReID also needs to use the trained CNN network to extract the global information of the vehicle images, and make feature similarity judgment in the embed layer; as the CNN network continues to develop, some excellent network structure has been widely used [6][21][15][22][17]; in the field of face recognition, the softmax Loss based on margin [26][25][4] increases the distance between intra-classes and compresses the distance between inter-classes, thereby greatly improving the performance of face recognition; in the field of pedestrian ReID, many methods [20][9][19] are also instructive to explore the vehicle ReID method;

In the vehicle ReID task, Liu et al [11]used the method similar to PCB in pedestrian ReID to cut the channels/width/height dimensions of the feature map, combined with the vehicle's feature information, supervised the network training, and concatenate the features output by

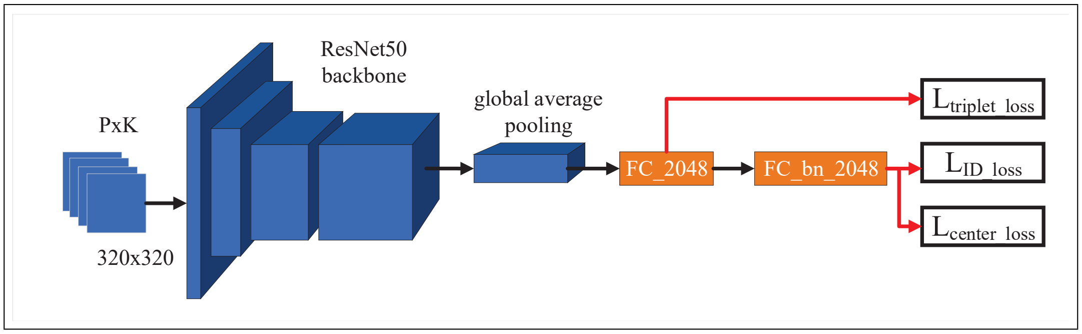

Figure 1. General architecture of our ResNet50 feature extractor. We removed the last stage down-sampling operation and use IBN to replace BN after the first 1x1 conv on the residual branch. F_{bn} refers to the features after BNNeck. The structure of other feature extractors is basically the same as the ResNet50 feature extractor, with only slight differences. For Densenet161 feature extractor, we keep the last spatial down-sampling operation in the network. For HRnetw18c feature extractor, We use BN instead of IBN after the first 1x1 conv on the residual branch. All feature extractors output F_{bn} feature as an apparent feature.

each branch network as the final feature; Wang et al [27] present 20 fixed key points for the vehicle, and combining the key points to strengthen the extracted special features to improve the representativeness of the feature; Chu et al [2] combined the orientation information of the vehicle, and processed the vehicles with the same orientation and the vehicles with different orientations separately to reduce the interference of the orientation on the results ; He and others [5] use the detection model to obtain areas such as car lights and annual inspection marks, and then enhance the characteristics of these areas to improve the characterization ability of the features; Lou and others [12] use GAN to generate some hard negative vehicles with the same brand but different details in the car, and then use these difficult samples to train the network; Sochor and others [16] use the vehicle's 3d box and perspective information to assist the network in extracting features.

Among other techniques, Luo et al [14]. Proposed some techniques when designing or training networks, including designing BNNeck, applying triplet loss and classification loss to different feature vectors, and strategies for learning rate warm up. The combination of ID loss, center loss and triplet loss, Random Erasing Augmentation and other methods are very effective. At the same time, the re-rank method proposed by Zhong et al [29]. Utilizes the mutual information between query and gallery to further optimize the performance of re-identification.

3. Proposed Approach

In this section, we introduce our proposed vehicle ReID method. Firstly, five feature extractors are applied with different model architectures to get robust features. Those feature extractors are trained with cross-entropy loss, triplet loss and center loss. Finally, a new re-ranking method by using query clustering and attribute constraint is adopted to refine the final results.

3.1. Feature Extraction

The based feature extractor uses ResNet50 network which is initialized with pre-trained parameters on ImageNet [3]. We removed the last stage down-sampling operation [13] because higher resolution feature maps enrich fine-grained features. IBN-Net [15] shows that combining Instance Normalization [24] and Batch Normalization (BN) [10] with an appropriate manner improves both learning and generalization capacities. For each block of ResNet50, we use IBN to replace BN layer after the first 1x1 conv on the residual branch. The 2048-dim features f are extracted by a fully-connected layer after ResNet50 network. Because of the target of ID loss and triplet loss are inconsistent in the embedding space, we apply a BNNeck [13] with zero bias after fully-connected layer to get the normalized feature f_{bn}. In the training stage, f are used to compute ID loss,and f_{bn} are used to compute triplet loss and center loss.

To get more distinguished features, we build five feature extractors based on the baseline by using different model architecture. we built a ResNet50 feature extractor as the backbone. Besides, We built a HRnetw18c [18] feature extractor and a Densenet161 [8] feature extractor. Then, an Attribute-Guided Network is trained for getting detailed features. In order to get a robust features against background noise, A Densenet161 feature extractor trained with background substitution is applied. Five feature extractors are trained separately. In the test phase, we concatenate all apparent features together as the final appearance signature, and use the concatenated features to compute the euclidean distances between query images and gallery images.

Random shrink: After cropped with bounding box, the average size of object images varies largely. Therefore, we adopt random shrink augment to improve the performance under this situation. For object with size larger than target resize shape, we generate a random number between 0.4

and 0.6 as the scaling factor. Then, we randomly scale down objects by the scaling factor with a probability of 0.5, and resize it to the input size. This operation can significantly improve the performance on small objects in test set.

Background substitution: In our experiments, training with simulation dataset helps to improve the performance of our model greatly. Inspired by this, we trained a PSP-net [28] to get the segmentation mask of vehicle identity and use it to split vehicle and background. Before training, background in images are randomly replaced from other input with a probability of 0.5.

Feature ensemble: To get more general and accurate features of vehicles in testing phase, we ensemble models by concatenating the features generated by ResNet50, Densenet161 with background substitution, HRnetw18c and Attribute-Guided Network(AGN) with corresponding weights as the appearance signature.

Loss function: Cross-entropy loss, triplet loss and center loss are used during training phase. Given an image, we denote y as truth label of ID and p_i as ID prediction logits of class i. We use feature f_{bn} to get ID prediction logits through softmax operation. The cross entropy loss is computed as:

$$L_{\text{ID}} = \sum_{i=1}^{N} -q_i \log(p_i) \left\{ \begin{array}{l} q_i = 0, y \neq i \\ q_i = 1, y = i \end{array} \right. . \tag{1}$$

Center loss are used to learn a center for deep features of each class and penalizes the distances between the deep features and their corresponding class.We use feature f_{bn} to compute center loss. The center loss function is formulated as:

$$L_{\text{center}} = \frac{1}{2} \sum_{j=1}^{B} \left\| \boldsymbol{f}_{t_j} - \boldsymbol{c}_{y_j} \right\|_2^2, \tag{2}$$

where y_j is the label of the jth image in a mini-batch. C_{y_j} denotes the y_j th class center of deep features. B is the number of batch size.

Simultaneously,we use feature f before BNNeck to compute triplet loss. The triplet loss function is formulated as:

$$L_{\text{triplet}} = [d_p - d_n + \alpha]_+ , \tag{3}$$

where d_p and d_n are the distances of positive pair and negative pair in the feature space. The α is the margin of triplet loss, and $[z]_+$ equals to max(z,0). In our experiments,we set α to 0.5.

Finally,the total loss for our feature extractor is:

$$L = L_{\text{ID}} + L_{\text{triplet}} + \beta L_{\text{center}}, \tag{4}$$

Where β is the weight of L_{center}, In our experiments, β is set to 0.0005.

3.2. Attribute-Guided Network

Attributes like car type, color, etc., are robust against the variations of vehicle appearance, which mainly caused by occlusion and different viewpoints. Inspired by [1], we proposed an Attribute-Guided network. As shown in Fig. 2. A general feature map F are generated by a ResNet50 backbone. Then, F is fed into a master branch to generate a ReID feature f_{id} and two attribute branches to get attribute-guided featuref_{type} and f_{color} for attribute classification task. The final concatenated feature f_c is regarded as the feature signature for vehicle ReID task.

In type and color classification branches, cross entropy loss is used for multi-task learning. And as same as feature extractors in 3.1, the loss L_{reid} is combined with cross entropy loss, center loss and triplet loss in ReID branch.

The overall loss function L_{attr} of Attribute-Guided Network can be formulated as:

$$L_{attr} = L_{reid} + \alpha L_{type} + \beta L_{color}, \tag{5}$$

where α and β are corresponding weights of f_{type} and f_{color}. In our experiment, we set both α and β to 1.

3.3. Post-processing

During the inference phase, two post-processing strategies are used to improve the ReID performance.

Attribute constraint: In order to better separate cars in different type and color in ReID task, We manually label type and color attributes for all Ids in track2 dataset and train a color classification network and a type classification network respectively to label the attributes of benchmark vehicles. And a constraint strategy is utilized on euclidean distance between query and gallery. After the constraint being adopted, the new distance d between query q_i and gallery g_j can be computed as:

$$d(i,j) = \begin{cases} d_{old}(i,j) + \delta_t & T_i \neq T_j \, and \, C_i = C_j \\ d_{old}(i,j) + \delta_c & T_i = T_j \, and \, C_i \neq C_j \\ d_{old}(i,j) + \delta_c + \delta_t & T_i = T_j \, and \, C_i = C_j \\ d_{old}(i,j) & otherwise \, , \end{cases} \tag{6}$$

where T_i and T_j are car types of q_i and g_j, C_i and C_j are colors of q_i and g_j, $d_{old}(i,j)$ means the euclidean distance between q_i and g_j, δ_t and δ_c are two punish-values.

Group Re-ranking: Due to different quality and pose variation of a query, getting the accurate representation is hard from only one image. To solve this problem, we group the queries with Euclidean distance lower than θ, and set the mean feature of the group as their representation. Meanwhile, galleries are clustered with same tracklet. Our re-ranking strategy is based on the query clusters and gallery clusters.

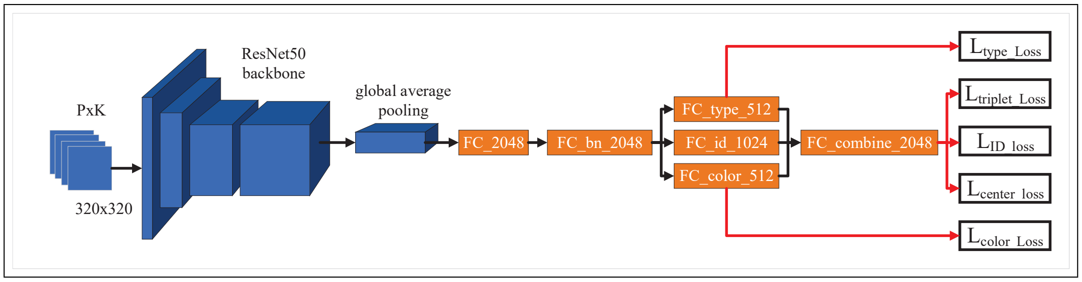

Figure 2. General architecture of our Attribute-Guided Network. This extractor structure is basically same as the ResNet50 feature extractor, but after BNNeck we add three new fully-connected layer for type, color and id attributes. finally, we concatenate three features after three fully-connected layer and add two cross-entropy losses to compute color loss and type loss. This feature extractor outputs f_{id} feature as an apparent feature.

Given ith query q_i with group mean feature c_i and jth gallery g_j with feature f_j. we regard g_j as the similar identity of q_i if the Euclidean distance between q_i and g_j is lower than θ, and then galleries with the same tracklet of g_j are inserted after g_j in the match sequence of q_i.

4. Experiments

4.1. Dataset

The AICITY2020 track2 dataset have two sub-data sets: AIC20 ReID dataset and AIC20 ReID Simulation dataset. AIC20 ReID dataset contains 56,277 images of 666 identities,which all come from multiple cameras placed at multiple intersections.The training set has 36,935 images which come from 333 vehicle identities. The test set consists of 18,290 images which belong to the other 333 identities. The rest of 1,052 images are used as query set. On average, each vehicle has 84.50 image signatures from 4.55 camera views. For the training set,each image has camera ID and vehicle ID ground truth. In addition, we also manually marked type ID, color ID, bounding boxes and segmentation ground truth for each image in the training set.AIC20 ReID Simulation dataset is generated by VehicleX, which is a publicly aviablable 3D engine. It has 192,150 images of 1,362 identities. Each image has camera ID,vehicle ID,color ID,type ID and the other ground truth. In our experiment, we add all images of them to the training set.

4.2. Implement Details

In training phase of all feature extractors, all input images are resized to 320 × 320 and padded with 10 pixels on image border, then images are randomly cropped to 320×320 again. Also, we apply random shrink and random erase with a probability of 0.5. Each feature extractor is trained for 120 epochs by Adam optimizer with weight decay of 5e-4. Warm-up learning strategy[13] are used in training, learning rate is linearly increased from 3.5e-5 to 3.5e-4 in the first 10 epochs, then divided by 10 at 40 epochs and again at 70 epochs. For ResNet50 and HRnet based feature extractors, we randomly select 64 identities and sample 4 images for each identity. For Densenet161 based feature extractors, we only select 24 identities.

Rank	Team ID	Score
1	73	0.8413
2	42	0.781
3	39	0.7322
4	36	0.6899
5	30	0.6684
6	**44(Ours)**	0.6683
7	72	0.6668
8	7	0.6561
9	46	0.6206
10	81	0.6191

Table 1. Performance Evaluation of Track2.

model	mAP
Res50	0.614
Res50+AGN	0.638
Res50+AGN+Dense161BS	0.657
Res50+AGN+Dense161BS+Dense161	0.662
Res50+AGN+Dense161BS+Dense161+HRnet	**0.668**

Table 2. results of model ensemble. Dense161BS mean Densenet161 feature extractor trained with background substitution augment.

4.3. Performance Evaluation of Challenge Contest

We report our challenge contest performance of the track2: City-Scale Multi-Camera Vehicle Re-Identification. In Track2, we rank number 6(team ID 44) among all the teams with the mAP of 0.6683, as can been in Table. 1. We show some ranking results in Fig. 3 produced by our method. And comparisons with model ensemble is shown in Table. 2.

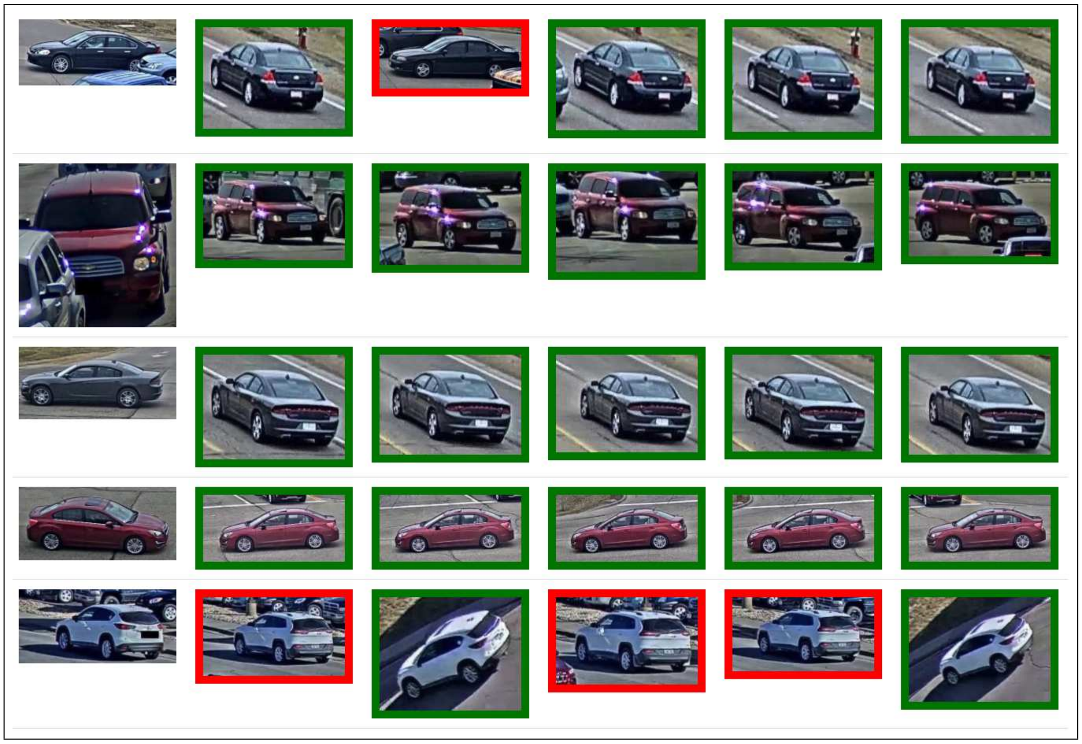

Figure 3. example ranking results of our method.

5. Conclusion

In this paper, we introduce a multi-feature extractor vehicle Re-ID system for learning a robust appearance feature for vehicle Re-ID. Based on a mature vehicle Re-ID system, we use pre-processing discussed above to get robust features. Then, we adopt the feature ensemble in our system to get powerful feature representations. After that, we use group re-ranking and attribute constraint to reduce the number of irrelevant gallery images. Finally, our proposed system rank number 6(team ID 44) among all the teams with the mAP of 0.6683 for City-Scale Multi-Camera Vehicle Re-Identification.

References

[1] Ejaz Ahmed, Michael Jones, and Tim K Marks. An improved deep learning architecture for person re-identification. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3908–3916, 2015.

[2] Ruihang Chu, Yifan Sun, Yadong Li, Zheng Liu, Chi Zhang, and Yichen Wei. Vehicle re-identification with viewpoint-aware metric learning. In Proceedings of the IEEE International Conference on Computer Vision, pages 8282–8291, 2019.

[3] 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.

[4] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4690–4699, 2019.

[5] 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.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, pages 770–778. IEEE Computer Society, 2016.

[7] Hung-Min Hsu, Tsung-Wei Huang, Gaoang Wang, Jiarui Cai, Zhichao Lei, and Jenq-Neng Hwang. Multi-camera tracking of vehicles based on deep features re-id and trajectory-based camera link models. In AI City Challenge Workshop, IEEE/CVF Computer Vision and Pattern Recognition (CVPR) Conference, Long Beach, California, 2019.

[8] 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.

[9] Houjing Huang, Wenjie Yang, Xiaotang Chen, Xin Zhao, Kaiqi Huang, Jinbin Lin, Guan Huang, and Dalong Du. Eanet: Enhancing alignment for cross-domain person re-identification. arXiv preprint arXiv:1812.11369, 2018.

[10] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167, 2015.

[11] Xiaobin Liu, Shiliang Zhang, Qingming Huang, and Wen Gao. Ram: a region-aware deep model for vehicle re-identification. In 2018 IEEE International Conference on Multimedia and Expo (ICME), pages 1–6. IEEE, 2018.

[12] Yihang Lou, Yan Bai, Jun Liu, Shiqi Wang, and Lingyu Duan. Veri-wild: A large dataset and a new method for vehicle re-identification in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3235–3243, 2019.

[13] 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.

[14] 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.

[15] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In Proceedings of the European Conference on Computer Vision (ECCV), pages 464–479, 2018.

[16] Jakub Sochor, Adam Herout, and Jiri Havel. Boxcars: 3d boxes as cnn input for improved fine-grained vehicle recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3006–3015, 2016.

[17] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5693–5703, 2019.

[18] Ke Sun, Yang Zhao, Borui Jiang, Tianheng Cheng, Bin Xiao, Dong Liu, Yadong Mu, Xinggang Wang, Wenyu Liu, and Jingdong Wang. High-resolution representations for labeling pixels and regions. arXiv preprint arXiv:1904.04514, 2019.

[19] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. arXiv preprint arXiv:2002.10857, 2020.

[20] Yifan Sun, Qin Xu, Yali Li, Chi Zhang, Yikang Li, Shengjin Wang, and Jian Sun. Perceive where to focus: Learning visibility-aware part-level features for partial person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 393–402, 2019.

[21] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In Thirty-first AAAI conference on artificial intelligence, 2017.

[22] Mingxing Tan and Quoc V Le. Efficientnet: Rethinking model scaling for convolutional neural networks. arXiv preprint arXiv:1905.11946, 2019.

[23] 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 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 8797–8806, 2019.

[24] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. arXiv preprint arXiv:1607.08022, 2016.

[25] Feng Wang, Jian Cheng, Weiyang Liu, and Haijun Liu. Additive margin softmax for face verification. IEEE Signal Processing Letters, 25(7):926–930, 2018.

[26] 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, pages 5265–5274, 2018.

[27] 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.

[28] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2881–2890, 2017.

[29] 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.