International Journal of Computer Vision (2021) 129:719–735
https://doi.org/10.1007/s11263-020-01402-2

Viewpoint and Scale Consistency Reinforcement for UAV Vehicle Re-Identification

Shangzhi Teng[1] · **Shiliang Zhang**[2] · **Qingming Huang**[1] · **Nicu Sebe**[3]

Received: 22 December 2019 / Accepted: 4 November 2020 / Published online: 24 November 2020

Abstract
This paper studies vehicle ReID in aerial videos taken by Unmanned Aerial Vehicles (UAVs). Compared with existing vehicle ReID tasks performed with fixed surveillance cameras, UAV vehicle ReID is still under-explored and could be more challenging, e.g., aerial videos have dynamic and complex backgrounds, different vehicles show similar appearance, and the same vehicle commonly show distinct viewpoints and scales. To facilitate the research on UAV vehicle ReID, this paper contributes a novel dataset called UAV-VeID. UAV-VeID contains 41,917 images of 4601 vehicles captured by UAVs, where each vehicle has multiple images taken from different viewpoints. UAV-VeID also includes a large-scale distractor set to encourage the research on efficient ReID schemes. Compared with existing vehicle ReID datasets, UAV-VeID exhibits substantial variances in viewpoints and scales of vehicles, thus requires more robust features. To alleviate the negative effects of those variances, this paper also proposes a viewpoint adversarial training strategy and a multi-scale consensus loss to promote the robustness and discriminative power of learned deep features. Extensive experiments on UAV-VeID show our approach outperforms recent vehicle ReID algorithms. Moreover, our method also achieves competitive performance compared with recent works on existing vehicle ReID datasets including VehicleID, VeRi-776 and VERI-Wild.

Keywords Vehicle re-identification · UAV · Viewpoint · Scale

1 Introduction

Vehicle re-identification (ReID) targets to match and identify query vehicles across different cameras. With the capability to accurately locate a specific vehicle, vehicle ReID is a fundamental vision task in smart traffic surveillance. Existing vehicle ReID works are mainly conducted with videos taken by traffic surveillance cameras. Although their number is already large in many cities, traffic surveillance cameras show limited coverage because of their fixed locations and limited viewpoints. In recent years, the Unmanned Aerial Vehicles (UAVs) technology has been substantially improved in terms of flight time, automatic control algorithm, and wireless image transmission, etc. For instance, the development of automatic wireless UAV charging stations has created an essential environment for continuous UAV operation. Compared with fixed surveillance cameras, mobile cameras on UAVs exhibit wider range of perspectives, as well as better mobility, flexibility and convenience. For instance, it is more efficient to collect traffic videos from a bird-eye view. Meanwhile, UAVs could actively track and record specific vehicles in both urban and highway scenarios. Due to those advantages, more and more cities in the world start to adopt UAVs in traffic surveillance. The advantages of UAV cameras, as well as the fast development of UAV technology, enable more efficient and active vehicle ReID. Besides that, UAV vehicle ReID algorithms can be integrated into existing ReID systems to achieve more intelligent traffic surveillance.

Communicated by Mei Chen.

✉ Shiliang Zhang
slzhang.jdl@pku.edu.cn

Shangzhi Teng
shangzhi.teng@vipl.ict.ac.cn

Qingming Huang
qmhuang@ucas.ac.cn

Nicu Sebe
sebe@disi.unitn.it

[1] University of Chinese Academy of Sciences, Beijing, China

[2] Department of Computer Science, Peking University, Beijing, China

[3] University of Trento, Trento, Italy

Vision tasks in UAV videos are drawing increasing attention from both industry and academia. Many UAV-related video datasets have been released. For example, Campus (Robicquet et al. 2016), CARPK (Hsieh et al. 2017) and UAV123 (Mueller et al. 2016) are collected by UAVs. These datasets define vision tasks of human trajectory analysis, object counting and tracking, respectively. Du et al. (2018) constructed an UAV dataset for object detection, single object tracking, and multiple object tracking. Avola et al. (2018) constructed an UAV dataset (UMCD) for mosaicking and change detection. A new UAV aerial video dataset (ManipalUAVid) for semantic segmentation was presented by Girisha et al. (2019). Zhu et al. (2020) and Zhu et al. (2018a) organized the UAV vision challenge workshops on object detection and tracking in ICCV 2019 and ECCV 2018, respectively. To the best of our knowledge, UAV vehicle ReID is still an under-explored task, and there is a lack of public dataset. To facilitate the research on this task, this work contributes a novel dataset called UAV vehicle re-identification (UAV-VeID). UAV-VeID contains 4601 vehicles, 41,917 annotated vehicle images and 16,850 query images. The covered scenarios involve complex backgrounds, various viewpoints, different illumination conditions and partial occlusions in the wild. We further add a distractor set composed of 300 K interference images to encourage the research on efficient ReID schemes. We believe this dataset is important because it is one of the first UAV vehicle ReID datasets. Meanwhile, UAV vehicle ReID is a key technique to achieve UAV vehicle tracking, searching and matching in smart traffic surveillance. It also has a very good potential to benefit other research on UAV video analysis, because different vision tasks on UAV videos face similar challenges like flexible viewpoints, scales, and backgrounds of objects.

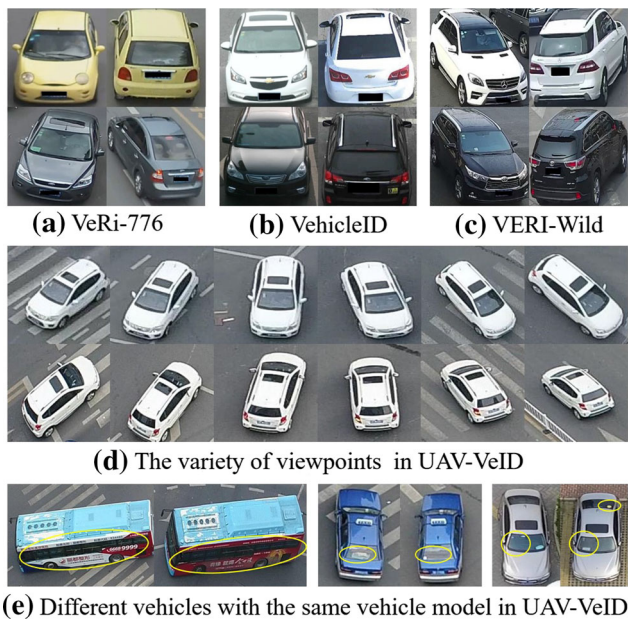


Fig. 1 Examples of vehicle images in vehicle ReID datasets, e.g., VeRi-776 (Liu et al. 2016d) in (**a**), VehicleID (Liu et al. 2016a) in (**b**), Veri-Wild (Lou et al. 2019) in (**c**), and UAV-VeID in (**d**, **e**), respectively

UAV could record vehicles from flexible viewpoints, altitudes, and under different illuminations. Figure 1 shows examples of vehicle images taken by surveillance cameras and UAV cameras, respectively. It is clear that, viewpoints of surveillance cameras are relatively fixed, e.g., vehicle images are generally taken from forward or rear views, where license plates regions are visible. Differently, in UAV-VeID the license plates are mostly invisible from bird-eye view. Note that, license plate regions on VeRi-776 (Liu et al. 2016d, c), VehicleID (Liu et al. 2016a) and VERI-Wild (Lou et al. 2019) are artificially occluded to encourage research on discriminative appearance feature learning. Figure 1d shows the variety of viewpoints in UAV-VeID. It is clear that, UAV vehicle ReID is more challenging because of flexible viewpoints, similar appearance and invisible details like headlight and maker-logo from bird-eye view. However, vehicle ReID in UAV videos is still feasible. Figure 1e illustrates some details among different vehicles with the same model in UAV-VeID. It can be observed that, those vehicles can be distinguished by the stickers on the body and decorations inside the vehicles. Therefore, UAV vehicle ReID involves many new challenges. It requires more discriminative and robust appearance features.

Existing methods on vehicle ReID commonly use Convolution Neural Network (CNN) to extract vehicle features and adopt distance metric learning to optimize feature distance (Liu et al. 2016c, d; Yan et al. 2017; Shen et al. 2017; Liu et al. 2018; Teng et al. 2018). Those algorithms perform well on existing datasets, but may be not optimal for UAV vehicle ReID. For instance, ignoring the variety of viewpoints during feature learning is not reasonable for UAV vehicle ReID, because vehicles under different viewpoints usually show substantially different visual appearances. As discussed in existing works (Chu et al. 2019), images of the same vehicle under different viewpoints may show larger distances than images of different vehicles under the same viewpoint. Some recent works (Wang et al. 2017; Zhou and Shao 2018; Khorramshahi et al. 2019; Chu et al. 2019; Teng et al. 2020) consider viewpoint cues to enhance the feature robustness. In addition, varied scales lead to different distributions in feature space (Tan et al. 2018), making multi-scale feature learning also important in ReID task. Therefore, it is desirable to fuse multi-scale features and enhance the robustness to scale variance. Some recent works (Teng et al. 2018; He et al. 2019a) fuse local and global features for vehicle ReID. Those works will be reviewed in Sect. 2

Feature spaces corresponding to different viewpoints or scales may distribute on different manifolds, which increase

the difficulty of matching the same vehicle in UAV videos. This motivates us to design a more effective feature learning strategy to reduce the distance between different feature manifolds. Specifically, we introduce an adversarial training algorithm and a multi-scale feature embedding module. The viewpoint adversarial training part consists of feature generator block, Gradient Reversal Layer (GRL) block, and a viewpoint discriminator, respectively. During adversarial training, the view discriminator tries to predict the viewpoint of the input vehicle from its feature vector. The feature generator tries to produce a viewpoint invariant feature and confuse the discriminator. The multi-scale embedding module consists of multi-scale feature extraction branch and multi-scale consensus loss computation. With the multi-scale consensus loss, the learned features under different scales become more comparable, which in-turn enhances the robustness to scale variances.

We conduct extensive experiments on UAV-VeID and compare our method against many recent ones. Comparisons show that our method achieves substantially better performance. We further test our method on existing vehicle ReID datasets, where it also shows competitive performance compared with the state-of-the art. This indicates that our method also works well on existing vehicle ReID datasets. The reason could be because that, viewpoint and scale variances also exist in existing datasets. We hence could conclude our contributions into two aspects: (1) we contribute a large-scale novel UAV-VeID dataset for UAV vehicle ReID. Compared with existing vehicle ReID datasets, UAV-VeID defines a more challenging and realistic vehicle ReID task. (2) We propose an easy-to-implement baseline algorithm to learn robust and discriminative visual feature for UAV vehicle ReID. This method shows promising performance on both UAV-VeID and existing vehicle ReID datasets.

The remainder of this paper is organized as follows. Section 2 reviews related works. Section 3 introduces the UAV-VeID dataset. Section 4 presents detailed descriptions to our algorithms. Section 5 summarizes experimental results, followed by conclusions in Sect. 6.

2 Related Work

This work is closely related with vehicle ReID, adversarial learning, and multi-scale feature embedding. This section briefly reviews those three categories of works, respectively.

2.1 Vehicle ReID

Recent object re-identification works commonly use metric loss, e.g., triplet loss (Schroff et al. 2015) to optimize the distance of learned deep features. Large margin softmax loss (Liu et al. 2016b) is a modified metric loss, which can generate more discriminative features by encouraging angular decision margin between classes. Center loss (Wei et al. 2016) is another metric loss, which enlarges the inter-class distance and decreases the intra-class distance. For vehicle ReID task, deep networks can learn Coupled Clusters Loss (CCL) (Liu et al. 2016a) to enhance the discriminative power of learned deep features. Another work Bai et al. (2018) proposes a group sensitive triplet embedding approach to model the inter-class dissimilarity as well as the intra-class invariance during neural network training. Most of existing algorithms directly optimize the distance between global features and do not consider the variance of viewpoints and scales. The dramatically varied visual appearances may degrade the effectiveness of metric learning algorithms. For instance, the CNN training could be hard to converge if most of the training samples are hard triplets, where the positive and anchor show dramatically different appearances and the negative shows similar appearance with the anchor.

Recently, regional attention learning strategies have been adopted in many vision tasks to enhance the feature or handle the misalignment issues (Chen et al. 2016; Lu et al. 2016; Zhu et al. 2018b; Li et al. 2018; Wei et al. 2017b). Many works use attention model or local cues in vehicle ReID. A recent work (Liu et al. 2018) proposes a Region-Aware deep Model to jointly learn deep features from both the global appearance and local regions. Teng et al. (2018) designed an attention module to refine the feature maps in CNN. Wang et al. (2017) pre-trained a region proposal module to produce the response maps of 20 vehicle key points, which are used to extract local features to enhance feature discriminative power. Another work (He et al. 2019a) integrates part and global information and achieves good performance on vehicle ReID. However, it needs an additional bounding box detection network for part localization, which is hard to generalize to side-view vehicle images. Most of part and attention based methods combine regional with global features to obtain multi-scale feature representations.

Besides distance metric optimization and local cues learning, many works enhance the feature robustness by utilizing viewpoint cues. For instance, Zhou and Shao (2018) proposed a viewpoint-aware attention model and focus on specific areas from different viewpoints. They design a conditional generative network to infer a multi-view feature representation from a single-view input. It boosts the robustness of visual feature to viewpoint variances, but is complicated and difficult to train. Chu et al. (2019) learned two metrics for similar viewpoints and different viewpoints in two feature spaces, respectively. During inference, viewpoint is firstly estimated and the corresponding metric is used. Khorramshahi et al. (2019) proposed an orientation conditioned key-point selection strategy, which is capable to localize and focus on the most informative parts of the vehicle. Teng et al. (2020) designed a multi-view branch network

to produce more discriminative viewpoint invariant feature. This paper alleviates the viewpoint variation influence with an adversarial learning approach, which is easier to implement. We also design a multi-scale embedding scheme to facilitate the multi-scale feature learning.

2.2 Adversarial Learning

Generative Adversarial Networks (GANs) has been widely used in recent years for feature learning and person or vehicle ReID (Wei et al. 2017a; Zhou and Shao 2018; Lou et al. 2019). For instance, Wei et al. (2017a) utilized GANs to generate training samples from the available training set. It relieves the expensive data annotations on new datasets and makes it easy to train person ReID systems for different testing domains. Some other works utilize the idea of adversarial learning to facilitate CNN training. Lou et al. (2019) design a feature distance adversary scheme to generate hard negative samples to facilitate ReID model training. Adversarial learning is also commonly used for domain adaption (Ganin and Lempitsky 2015; Ganin et al. 2016; Tzeng et al. 2017; Pei et al. 2018; Long et al. 2018). Previous works (Ganin and Lempitsky 2015; Ganin et al. 2016) propose a Gradient Reversal Layer (GRL) to embed domain adaptation into the representation learning procedure. The GRL does not affect the forward propagation, but reverses the gradients during the back-propagation. A Multi-Adversarial Domain Adaptation (MADA) (Pei et al. 2018) approach is proposed for the fine-grained alignment of different data distributions based on multiple domain discriminators. Long et al. (2018) presented conditional domain adversarial networks to exploit discriminative information to assist adversarial adaptation. In this work, we employ adversarial learning to learn visual features robust to viewpoint changes. To the best of our knowledge, this is an early work that use adversarial adaptation for feature learning in vehicle ReID.

2.3 Multi-scale Embedding

Multi-scale visual cues are important for vision tasks such as classification (Huang and Chen 2018), object detection (Li et al. 2019), semantic segmentation (He et al. 2019b), crowd counting (Liu et al. 2019a) and person ReID (Qian et al. 2017; Chang et al. 2018). Some works build image pyramid or feature pyramid to learn multi-scale cues. Chang et al. (2018); Qian et al. (2017) designed multi-stream building blocks to learn multi-scale features for person ReID. Zhou et al. (2019) designed a residual block composed of multiple convolutional feature streams for omni-scale feature learning. This work employs dilated convolution in a multi-branch architecture to obtain multi-scale features. The dilation operation alleviates the expensive scale pyramid construction and effectively learns neurons with multi-scale receptive fields. As shown in our experiments, this design effectively learns multi-scale features and facilitates the learning of scale invariance features, which is important for boost the ReID accuracy in UAV-VeID.

3 UAV-VeID Dataset

This section first reviews existing vehicle ReID datasets, then proceeds to introduce the UAV-VeID dataset.

3.1 Existing Datasets

Current vehicle ReID algorithms are mainly tested on two benchmark datasets, i.e., VeRi-776 (Liu et al. 2016d) and VehicleID (Liu et al. 2016a), respectively. A new vehicle ReID dataset VERI-Wild (Lou et al. 2019) is also proposed in 2019. VeRi-776, VehicleID and VERI-Wild are all captured by traffic surveillance cameras equipped on urban roads.

3.1.1 VeRi-776

Liu et al. (2016d, c) is collected from the traffic surveillance scenarios, with 51,035 images of 776 vehicles in total. It is split into 576 vehicles with 37,778 images for training and 200 vehicles with 11,579 images for testing. 1678 images selected from the test set are used as query images. Most of images are captured from the forward and backward viewpoints. A small portion of vehicle images in VeRi-776 are captured from the side-view.

3.1.2 VehicleID

Liu et al. 2016a consists of 26,267 vehicles and 221,763 images in total. It provides a training set with 100,182 images from 13,164 vehicles and a test set with 20,038 images from 2400 vehicles. Vehicle images in VehicleID are either captured from the forward or the backward views.

3.1.3 VERI-Wild

Lou et al. (2019) is a large-scale vehicle ReID dataset containing 416,314 vehicle images of 40,671 identities. It is captured by a traffic surveillance camera system consisting of 174 cameras across one month under unconstrained scenarios. The YOLO-v2 (Redmon and Farhadi 2017a) is used to detect vehicle bounding boxes. It is randomly divided into two parts for training and testing, i.e., 30,671 vehicles with 277,797 images for training and three subsets for testing.

Figure 1 illustrates several images from VeRi-776, VehicleID and VERI-Wild, respectively. Although current algorithms have achieved high accuracy on those datasets, vehicle ReID system might suffer from the limited coverage and

flexility of fixed traffic surveillance cameras. Moreover, artificially occluding license plate regions makes existing datasets different from the ones in real scenarios. UAV videos have potential be applied for more realistic and active vehicle ReID. The following parts proceed to introduce the UAV-VeID dataset.

3.2 UAV-VeID

3.2.1 Data Collection

We simulate real scenarios as much as possible during UAV videos collection. Specifically, UAV videos are collected from different locations with distinct backgrounds and lighting conditions, e.g., including highways, urban road intersections, and parking lots, etc., as shown in Fig. 2a. For vehicles at parking lots, we adopt various UAV sport modes such as cruising and rotating to record vehicles. This strategy introduces viewpoint and scale changes, as well as partial occlusions to images of the same vehicle. For moving vehicles, we use two UAVs to simultaneously shoot videos from different viewpoints and heights. This strategy introduces viewpoint, scale, and background changes. The flying height of UAVs ranges from 15 to 60 meters, leading to different scales of vehicle images. The vertical angle of UAV camera ranges from 40 to 80 degrees, which leads to different viewpoints of vehicle images. The videos are recorded at 30 frames per second (fps), with the resolution of 2704×1520 pixels and 4096×2160 pixels, respectively. The UAV-VeID is constructed from 80 video sequences selected from raw UAV videos.

3.2.2 Annotation

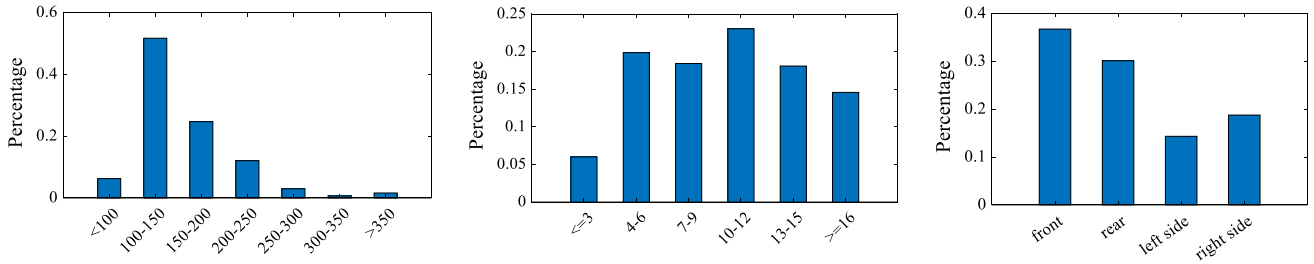
We annotate vehicles from collected videos to construct the UAV-VeID. In each video clip, 1 video frame is sampled

(a) Dataset Capture Conditions

(b) Samples of UAV-VeID

(c) Distractor Images

Fig. 2 Sample images from the UAV-VeID dataset. **a** The locations for video collection, including urban road intersections, highways, and parking lots. **b** Examples of annotated vehicle images with district viewpoints and scales. **c** Examples of distractor images. UAV-VeID involves variances of viewpoint, illumination, backgrounds, and scales, hence could be a challenging dataset for vehicle ReID



(a) Bounding Box Width (pixels) **(b)** Image Number Per Vehicle **(c)** Viewpoints Category

Fig. 3 Statistics on UAV-VeID dataset. **a–c** The statistics on image resolution, number of images per vehicle, as well as the viewpoint variance on UAV-VeID, respectively

Table 1 Comparison between UAV-VeID and existing vehicle ReID datasets

Dataset	VeRi-776	VehicleID	VERI-Wild	UAV-VeID
Identities	776	26,267	40,671	4601
Images	51,035	221,763	416,314	41,917
Distractors	0	0	0	300 K
Cameras	Fixed	Fixed	Fixed	Mobile UAV
Views	3	2	Flexible	Flexible
Occlusion	No	No	Yes	Yes
Background	Fixed	Fixed	Flexible	Flexible
Weather	Fixed	Fixed	Flexible	Flexible
Year	2016	2016	2019	–

every one second to construct a video frame dataset. The dataset annotation is hence conducted based on those sample video frames. When labeling the moving vehicles, we first align videos from two UAVs in time, then match the same vehicles from those two videos. To finish the vehicle annotation, 6 domain experts are involved to manually locate and annotate the identities of vehicles from each video frame. The data annotation procedure takes 1000 man-hours and finally results in a dataset containing 41,917 vehicle bounding boxes of 4601 vehicles. Each vehicle is annotated by at least two bounding boxes. The statistics of UAV-VeID dataset are illustrated in Fig. 3. For instance, Fig. 3b shows that, most of annotated vehicles have more than 3 bounding boxes.

In the setting of person and vehicle ReID tasks, training, validation, and testing set do not share the same IDs. In other words, IDs in the testing set are not included in the training and validation sets. This setting makes the ReID task more challenging and realistic, i.e., the trained ReID model should learn robust features and strong generalization ability to work on unseen test samples. We follow this setting and randomly divide UAV-VeID into three parts for training, validation and testing, respectively. The training set contains 18,709 bounding boxes of 1797 identities, the validation set contains 4150 bounding boxes of 596 identities and the test set contains 19,058 bounding boxes of 2208 identities, respectively. For validation set and testing set in UAV-VeID, we randomly select one image of each vehicle and put it into the gallery set. Other images are used as queries. We compare UAV-VeID with existing datasets in Table 1. Figure 3 shows statistics of bounding box size, number of images per vehicle, as well as viewpoint variance on UAV-VeID. Compared with existing datasets, UAV-VeID presents the following new properties:

- *Flexible viewpoint and scale* Taken by UAV cameras from the air, vehicle images in UAV-VeID present more flexible viewpoints, orientations, and scales, as shown in the Fig. 2b. Figure 3c further shows the statistics of viewpoint variances in UAV-VeID. Vehicles show distinct appearances under different viewpoints, making UAV-VeID more challenging than existing datasets.
- *More realistic task* Different from existing vehicle ReID datasets, vehicle images in UAV-VeID are collected under unconstrained conditions and are not artificially modified. It hence defines a more realistic vehicle ReID task.
- *A large distractor set* We introduce a large distractor set consisting of falsely detected bounding boxes, as well as vehicle images not belonging to the 4601 annotated identities. Sample images are shown in Fig. 2c. Adding the distractor set encourages the research on more efficient vehicle ReID.
- *Introduction of a validation set* We randomly divide UAV-VeID into training set, validation set, and testing set, respectively. Tuning algorithms on validation set rather than the test set could make more reasonable experimental comparisons.

The following part proceeds to introduce our proposed methods to learn robust features for UAV vehicle ReID.

4 Methodology

4.1 Formulation

To extract visual features robust to viewpoint variance, this paper defines different viewpoints as distinct domains and utilizes adversarial learning for feature training. Domain adversarial learning has been successfully applied in transfer learning (Ganin and Lempitsky 2014, 2015; Ganin et al. 2016; Tzeng et al. 2017; Pei et al. 2018; Long et al. 2018) to reduce the distribution shift between the source and target domains. The adversarial learning can be regarded as a two-player game, where the first player is the domain discriminator trained to distinguish the source domain from the target domain, and the second player is the feature extractor fine-tuned to confuse the domain discriminator. Those two players are simultaneously updated, leading to a stronger feature extractor robust to domain gap.

View adversarial learning involves three models to be trained, i.e., feature extractor G, viewpoint discriminator D_v, and label predictor D_y. We use θ_f, θ_v, and θ_y to denote their parameters, respectively. To extract view-invariant features f, parameters θ_f of G are learned by maximizing the loss of viewpoint discriminator D_v and minimizing the loss of label predictor D_y. While parameters θ_v of D_v are learned by minimizing the loss of the viewpoint discriminator. The objective of view adversarial learning can be formulated as the combination of two functions, i.e.,

$$\begin{aligned} E(\theta_f, \theta_y, \theta_v) &= l(D_y(G(x;\theta_f);\theta_y), y) \\ &\quad - \lambda l(D_v(G(x;\theta_f);\theta_v), v) \\ &= L_y(\theta_f, \theta_y) - \lambda L_v(\theta_f, \theta_v), \end{aligned} \tag{1}$$

where x is the input image, y is the annotated classification label, and v is the viewpoint label, $l(\cdot)$ computes the loss between predicted and groundtruth labels. L_y and L_v are loss functions for label prediction and viewpoint classification, respectively. λ is a trade-off parameter.

During the training procedure, optimized network parameters $\hat{\theta}_f, \hat{\theta}_y, \hat{\theta}_v$ can be obtained as,

$$\begin{aligned} (\hat{\theta}_f, \hat{\theta}_y) &= \arg\min_{\theta_f, \theta_y} E(\theta_f, \theta_y, \hat{\theta}_v), \\ (\hat{\theta}_v) &= \arg\max_{\theta_v} E(\hat{\theta}_f, \hat{\theta}_y, \theta_v). \end{aligned} \tag{2}$$

The standard stochastic gradient solvers can be updated as,

$$\theta_f = \theta_f - \gamma \left(\frac{\partial L_y}{\partial \theta_f} - \lambda \frac{\partial L_v}{\partial \theta_f} \right), \tag{3}$$

$$\theta_y = \theta_y - \gamma \frac{\partial L_y}{\partial \theta_y}, \tag{4}$$

$$\theta_v = \theta_v - \gamma \frac{\partial L_v}{\partial \theta_v}, \tag{5}$$

where γ is the learning rate. The $-\lambda$ in Eq. (3) represents the Gradient Reverse Layer (GRL), which efficiently ensures the learned features to be robust to viewpoint variances.

To learn scale invariant features, one possible solution is to enforce features corresponding to different scales perform similarly in vehicle ReID. We hence extract features corresponding to different scales and compute a Multi-Scale Consensus Loss (MSCL) among them. Multi-scale features can be extracted from inputs with different scales using the same extractor, or from the same input using multi-scale feature extractors. Instead of taking multiple images as input, our framework generates multi-scale features through parallel branches with different dilation rates. Dilated convolution with dilation rate d inserts $d-1$ zeros between consecutive filters. This operation efficiently enlarges the receptive field without increasing the number of parameters and computations.

We hence implement multiple branches with different dilation rates to extract multi-scale features from the same input image. We denote features extracted with dilation rate $d > 1$ as f^*. With f and f^*, the MSCL is defined as,

$$L_s = \sum_{b=1}^{B-1} \sum_{m=1}^{Dim} |f(m) - f_b^*(m)|_2, \tag{6}$$

where B is the number of branches computed with dilation rate $d \geq 1$, Dim is the dimension of feature f and f^*, $|\cdot|_2$ computes the L_2 distance.

It is easy to infer than, L_s would be minimized, if features with different scales are similar for the same vehicle. This essentially ensures features from each branch to be robust to scale variances. In other words, f and f^* learn from each other to assist scale-invariant feature learning. The following parts present how we implement L_y, L_v, and L_s with a CNN.

4.2 Implementation

To implement and optimize the formulation in Eq. (2) and Eq. (6), we propose a viewpoint and scale consistency reinforcement framework illustrated in Fig. 4. As shown in the figure, our framework consists of a main branch to learn the vehicle feature f. Several side branches with different dilation rates are implemented to learn feature f^*. This part introduces the implementations to those components.

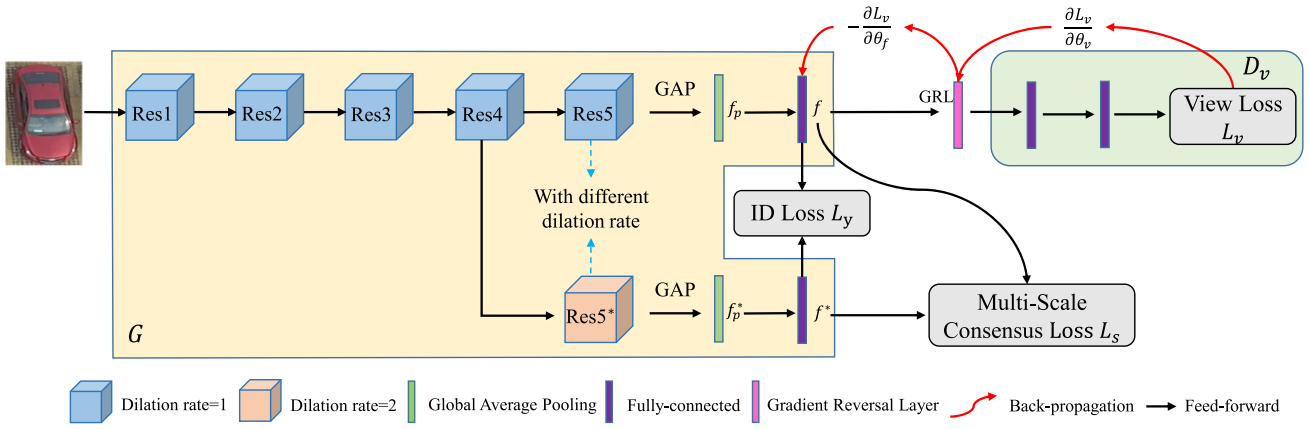


Fig. 4 Illustration of the proposed framework. For an input vehicle image, the feature extractor G generates feature vector f. Multiple side branches with dilation rate $d > 1$ are implemented to learn multi-scale features f^*. View discriminator D_v is connected to f by the GRL to enhance its robustness to viewpoint variance. MSCL L_s is computed between f and f^* to enhance the scale invariance. After training, the side branches are discarded and the feature f is used for vehicle ReID

4.2.1 Scale Invariance Training

We use ResNet50 (He et al. 2016a) as the backbone network. Different from the original ResNet50, two fully-connected layers are inserted after the Global Average Pooling (GAP) layer to learn the feature f. The first fully-connected layer with 512 neurons plays the role of feature dimension reduction. The second fully-connected layer executes vehicle identity categorization. For the multi-scale feature learning, we implement multi-branches, i.e., B branches, to learn multi-scale representations. Each branch shares the same structure with the original convolution branch, but has different dilation rates at the final convolutional block. For ResNet50, each residual block consists of three convolutions with kernel size 1×1, 3×3, and 1×1, respectively, where we set different dilation rates for the 3×3 convolution kernel. As shown in Fig. 4, to enhance the efficiency, we only change the dilation rate in the res5 stage. More details could be found in Sect. 5.3. In Fig. 4, the main branch produces f and side branches produce multi-scale features f^*, allowing for the computation of MSCL L_s with Eq. (6).

4.2.2 Viewpoint Adversarial Training

The view adversarial training is achieved by connecting a view discriminator to the 512-D feature f through a Gradient Reversal Layer (GRL). We implement the view discriminators with two fully connected layers. During adversarial training, the view discriminator, i.e., D_v in Fig. 4, tries to predict the viewpoint of the input vehicle from its feature vector. The learned feature f by G should be invariant to viewpoint changes to confuse the D_v. The GRL is the key to achieve adversarial training. During the forward propagation, GRL acts as an identity transform. During the back propagation in adversarial training, GRL takes the gradient from the subsequent layers, multiplies it by $-\lambda$ and passes it to the preceding layers. This operation effectively achieves adversarial training and has no parameters to learn. It is also simple to implement GRL using existing deep learning packages.

During viewpoint adversarial training, the viewpoint classification loss L_v is defined as,

$$L_v = -\sum_{i=1}^{n_v} [v(i) \cdot log(\hat{v}) + (1 - v(i)) \cdot log(1 - \hat{v})], \quad (7)$$

where $\hat{v}$ and v refer to the predicted viewpoint probability value and viewpoint label, respectively. n_v denotes the number of viewpoint categories in the training set.

4.2.3 Training Loss

We train the framework in Fig. 4 with three loss functions. The vehicle ID discriminator predicts the vehicle ID label from the learned feature f and f^*, and enforces images from the same vehicle have more similar features than images of different vehicles. We hence define the ID label prediction loss L_y as:

$$L_y = L_c + \lambda_1 L_t, \quad (8)$$

where L_c is the softmax cross entropy loss computed with the ground truth ID label, and L_t is the triplet loss computed on image triplets. λ_1 is a weighting parameter.

The ID classification loss L_c is defined as:

$$L_c(f) = -\sum_{j=1}^{n_{id}} y(j) \cdot log\left(\frac{exp(\omega_j^\top f)}{\sum_{k=1}^{n_{id}} exp(\omega_k^\top f)}\right), \quad (9)$$

where f and y refer to the extracted feature vector and vehicle ID label. n_{id} denotes the number of vehicle categories in the training set, and ω_k denotes the classifier parameters of the kth category. Note that, the ground truth label y is an one-hot vector.

Triplet loss enforces one feature to be closer to another feature from the same vehicle, than to a feature from any other vehicles. The triplet loss is defined as,

$$L_t(f^{(a)}, f^{(p)}, f^{(n)}) = max(0, |f^{(a)}, f^{(p)}|_2 - |f^{(a)}, f^{(n)}|_2 + m), \quad (10)$$

where $|\cdot|_2$ represents the L_2 distance, superscripts a, p, and n denote the anchor sample, positive sample and negative sample, respectively. m is a constant threshold value.

The final training loss function is defined as:

$$L = L_c + \lambda_1 L_t + \lambda_2 L_v + \lambda_3 L_s, \quad (11)$$

where L_s is multi-scale consensus loss defined in Eq. (6), and λ_1, λ_2, and λ_3 are the weighting parameters.

4.2.4 Inference

After network training, we only keep the feature extractor G for vehicle feature extraction and discard the side branches. As shown in Fig. 4, G takes the global images as input and outputs 512-D features, i.e., f. We use f as the final vehicle feature and apply linear search and Euclidean Distance for vehicle ReID.

5 Experiments

5.1 Experimental Settings

We use the mean Average Precision (mAP) and Cumulative Match Curve (CMC) to evaluate ReID performance. UAV-VeID, VeRi-776, VehicleID, and VERI-Wild are used for experiments. We follow the standard experimental settings provided by VeRi-776, VehicleID, and VERI-Wild, respectively. For the VehicleID dataset, only the large query set is evaluated since it has 2400 identities, and it is the most challenging test set. During testing, one image is randomly selected from one identity to generate a gallery set with 2400 images, the remaining 17,638 images are all used as query (probe) images. The random selection process was repeated for 10 times to obtain an average CMC result. For the VeRi-776 dataset, we select 11,579 images as gallery set and the remaining 1678 images are selected as probe images. The selection is the same as Liu et al. (2016d).

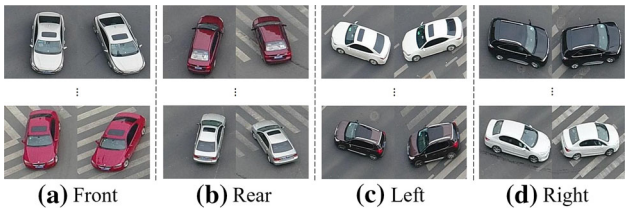


Fig. 5 Examples of four defined viewpoint categories in UAV-VeID dataset

5.2 Implementation Details

5.2.1 Viewpoint Annotation

Our view adversarial training module requires viewpoint labels to train the view discriminator. Most of vehicles taken by traffic surveillance cameras show front or rear views. Differently, vehicles in UAV may show continuous views, making it is hard to define and annotate all viewpoints. We hence simply define four typical viewpoints for UAV-VeID to verify the effectiveness of our method. We annotate all the training vehicle images of UAV-VeID with four viewpoint labels, i.e., front, rear, left, and right, respectively. Figure 5 shows examples from those four viewpoint categories in UAV-VeID. Other vehicle ReID datasets like VehicleID (Liu et al. 2016a) and VeRi-776 (Liu et al. 2016d) do not provide viewpoint annotations. Vehicles are captured from either front or rear viewpoint in VehicleID, so we define front and rear as the viewpoint labels. As for Veri-776 and VERI-Wild, the number of side view images is relatively small. We combine the left and right views as the same category (side), and then label the viewpoint of each image as front, side, or rear, respectively. To train the viewpoint classifier, we annotate all vehicle images in VeRi-776 with front, rear and side labels. Then we use annotated images to train a three-class viewpoint classifier for VeRi-776 and VERI-Wild, respectively. A two-class classifier for VehicleID is also trained using front and rear view labels on VeRi-776. Because viewpoint recognition is not a challenging task, we implement our viewpoint classifier based on VGG_CNN_M_1024 (Chatfield et al. 2014).

5.2.2 Training Details

We use ResNet50 as the backbone for vehicle feature learning. We remove the last spatial down-sampling operation in the backbone network to increase the size of the feature map (Sun et al. 2018). We initialize the ResNet50 with pre-trained parameters on ImageNet and change the dimension of the output layer to the number of identities in the training set. We randomly sample P vehicles and K images of per vehicle to constitute a training batch. Finally the batch size equals to $P \times K$. In this paper, we set $P = 8$ and $K = 4$. We resize each image into 224×224, then perform random erasing

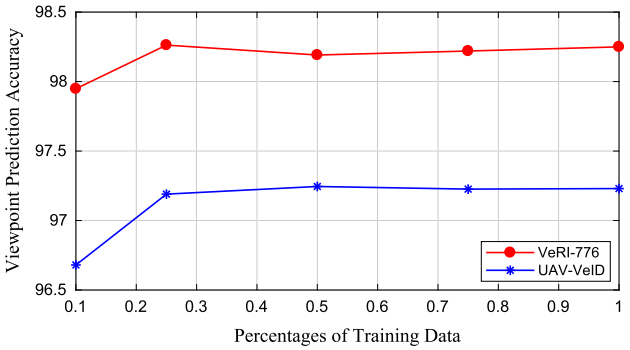


Fig. 6 The viewpoint prediction accuracy evaluated on different percentages of training data. The red line denotes the results on VeRi-776 and the blue line denotes the results on UAV-VeID (Color figure online)

augmentation on VeRi-776, VehicleID and VERI-Wild. The margin m of triplet loss is set to be 0.3. The initial learning rate is set to 0.01 and is decreased by 0.1 at the 30th epoch, 60th epoch and 90th epoch respectively. Totally, there are 120 training epochs. The weight decay factor is set to 0.0003. The momentum is set as 0.9. The loss weights in Eq. (11) are set to $\lambda_1 = 1.0, \lambda_2 = 1.0, \lambda_3 = 1.0$, respectively.

5.3 Ablation Studies

5.3.1 Accuracy of Viewpoint Prediction

Viewpoint prediction is important for our view adversarial learning. We hence first conduct experiments to evaluate the accuracy of viewpoint prediction. For UAV-VeID and VeRi-776, we generate training and testing sets based on our annotated viewpoint labels for viewpoint classification, respectively. For UAV-VeID, the training set contains 18,709 images, and the rest 3742 images are used as the test set. For VeRi-776, 30,222 images are used as training set and the rest 7556 images are used as the test set. Figure 6 summarizes the viewpoint classification accuracy using different portions of training images for viewpoint classifier training. It is clear that, using 25% training images could get reasonably good label prediction accuracy, e.g., 98.29% on VeRI-776. Increasing the number of training images does not bring substantial performance gains. This could be because viewpoint classification is an easy task, i.e., only has 2–4 classes, thus does not require a large training set. Therefore, our strategy does not require too many efforts for viewpoint annotation.

Table 3 Performance of different viewpoint categories for the viewpoint adversarial training on UAV-VeID validation set

Methods	r=1	r=5	r=10	mAP
Baseline	83.52	95.97	98.13	89.01
3-View adversarial	85.71	97.68	98.76	90.97
4-View adversarial	**86.05**	**98.03**	**99.01**	**91.29**

Bold denotes the best performance

5.3.2 Validity of Scale Invariance Training

Our scale invariance training involves an important parameter, i.e., the branch number B. This part evaluates the validity of our scale invariance training module, as well as the effects of B. We test different variants of our scale invariance training strategy and summarize the results in Table 2, where (a) denotes the baseline.

Table 2(b–e) summarizes the results of using two branches with dilation rates 1 and 2, respectively. The comparison between Table 2(d, e) shows the validity of MSCL, i.e., training with MSCL leads to a better f than directly fusing f and f^* together. This shows that MSCL effectively boosts the feature performance. The comparison between Table 2(c, e) shows that it is reasonable to learn different parameters for different branches, rather than sharing the same parame-

Table 2 Validity of our scale invariance training strategy

Dataset / Methods	Dilation rate	Weight-sharing	MSCL	Dim	UAV-VeID Validation		VeRi-776	
					r=1	mAP	r=1	mAP
(a)	1	–	–	512	83.52	89.01	88.25	63.01
(b)	1, 2	✓	–	1024	85.73	90.29	88.51	64.33
(c)	1, 2	✓	✓	512	86.21	91.06	88.72	64.56
(d)	1, 2	–	–	1024	88.02	92.36	90.25	66.36
(e)	1, 2	–	✓	512	89.32	93.03	91.78	68.95
(f)	1, 2, 3	–	–	1536	89.41	93.07	91.35	68.06
(g)	1, 2, 3	–	✓	512	**91.32**	**94.76**	**92.96**	**72.18**
(h)	1, 2, 3, 4	–	✓	512	**91.33**	**94.28**	**93.01**	**72.29**

Bold denotes the best performance

"Dilation Rate" shows the branch number B, and the dilations rates in each branch. "Weight-Sharing" denote sharing the same parameters among different side branches. "MSCL" denotes using the trained f for ReID. Without "MSCL" denotes fusing features from multiple branches for ReID. Each branch produces a 512-dim feature

ters. Similar conclusion can be drawn from the comparison between Table 2(b, d).

We further introduce more branches and summarize the results in Table 2(f–h). It is clear that, more branches is beneficial for the performance improvement. As shown in Table 2(e, g), introducing an extra branch boosts the rank-1 accuracy from 89.32 to 91.32% on UAV-VeID validation set. It is also interesting to observe from Table 2(f) that, directly fusing features from three branches leads to a higher dimensional feature, but could not substantially improve the performance. Compared with Table 2(f, g) achieves better performance with a lower-dimensional feature. This further shows the validity of MSCL, which enhances feature f with multi-scale features f^*.

Table 2 also shows that, introducing too many branches does not bring substantial performance gains, e.g., 3 branches perform similarity with 4 branches. Therefore, we set the branch number B as 3, and use the setting in Table 2(g) to implement our scale invariance training in the following experiments. The adopted setting achieves reasonably good performance on both UAV-VeID and VeRi-776.

5.3.3 Validity of Viewpoint Adversarial Training

This part further tests the validity of viewpoint adversarial training. We test the performance of different viewpoint categories for the viewpoint adversarial training on UAV-VeID validation set. In Table 3, the Baseline is trained only using cross entropy loss L_c and triplet loss L_t, without any viewpoint information. 3-View Adversarial denotes our proposed viewpoint adversarial training with 3 viewpoints, i.e., front, rear and side. 4-View Adversarial denotes our proposed viewpoint adversarial training with 4 viewpoints, i.e., front, rear, left and right. It is clear that, 3-Viewpoint outperforms the Baseline, e.g., improves baseline rank1 accuracy from 83.52 to 85.71%, but is still lower than the 86.05% of 4-Viewpoint. Because the same vehicle might have different markers on the left and right sides, it is more reasonable to divide side viewpoint as two left and right viewpoints. In Table 4(a, b), we summarize the performance without and with viewpoint adversarial training on UAV-VeID, VeRi-776, and VehicleID. Table 4(b) adds the view adversarial learning module to the baseline. Based on the viewpoint attribute, we intend to learn a viewpoint invariant feature representation. The comparison clearly shows that, view adversarial learning effectively boosts the performance on UAV-VeID, VeRi-776, and VehicleID. Therefore, it is necessary to make use of viewpoint information for vehicle feature representation learning.

Table 4(c) shows the performance of using scale invariance training. The performance of combining viewpoint adversarial training and scale invariance training is summarized in Table 4(d). Table 4(b–d) show that, both of our two training strategies are important for the performance gains, and their combination, i.e., the VSCR leads to the best performance. For example, VSCR achieves rank-1 accuracy of 91.98% on UAV-VeID validation set, significantly better than the 83.52% of baseline. Our method also performs well on existing datasets, e.g., Table 4b, c achieve rank-1 of 91.35% and 92.96% on VeRi-776, better than the baseline 88.25%. VSCR also achieves the best performance on existing vehicle ReID datasets VeRi-776 and VehicleID.

Table 5 Comparison with recent works on UAV-VeID test set

Methods	r = 1	r = 5	r = 10
VGG_CNN_M (Chatfield et al. 2014)	28.34	39.27	43.48
Siamese-Visual (Shen et al. 2017)	25.98	41.98	50.61
RAM (Liu et al. 2018)	45.26	59.35	64.07
SCAN (Teng et al. 2018)	40.49	53.74	60.55
GoogLeNet (Szegedy et al. 2015b)	45.23	64.88	70.38
CN-Nets (Yao et al. 2017)	55.91	76.54	82.46
VSCR (Ours)	**70.59**	**88.33**	**92.51**

Bold denotes the best performance

5.4 Comparison with Recent Works

5.4.1 UAV-VeID

Table 5 compares our method with recent ReID and fine-grained feature learning method on UAV-VeID test set. Compared methods include Siamese-Visual (Shen et al. 2017), CN-Nets (Yao et al. 2017), RAM (Liu et al. 2018) and SCAN (Teng et al. 2018), etc. We implement those

Table 4 Performance of different variants of our approach on UAV-VeID, VeRi-776, and VehicleID

Dataset	UAV-VeID Validation				VeRi-776				VehicleID			
Methods	r = 1	r = 5	r = 10	mAP	r = 1	r = 5	r = 10	mAP	r = 1	r = 5	r = 10	mAP
(a) Baseline	83.52	95.97	98.13	89.01	88.25	94.69	96.55	63.01	66.96	77.39	82.73	71.27
(b) View adversarial	86.05	98.03	99.01	91.29	91.35	96.18	97.15	68.35	69.94	81.02	85.01	75.76
(c) Multi-scale	91.32	98.63	99.16	94.76	92.96	96.89	97.86	72.18	73.49	85.16	89.28	77.71
(d) VSCR	**91.98**	**98.86**	**99.37**	**95.21**	**94.11**	**97.85**	**98.56**	**75.53**	**74.58**	**87.12**	**92.09**	**78.78**

Bold denotes the best performance

VSCR (Viewpoint-Scale Consistency Reinforcement) denotes the proposed entire framework in Fig. 4

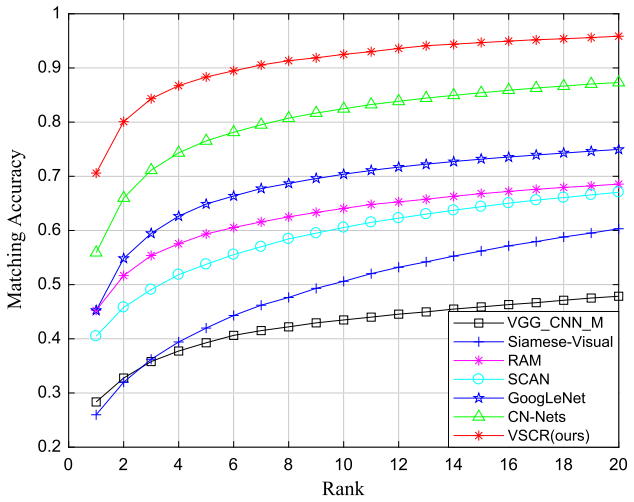


Fig. 7 CMC curves of compared methods on UAV-VeID

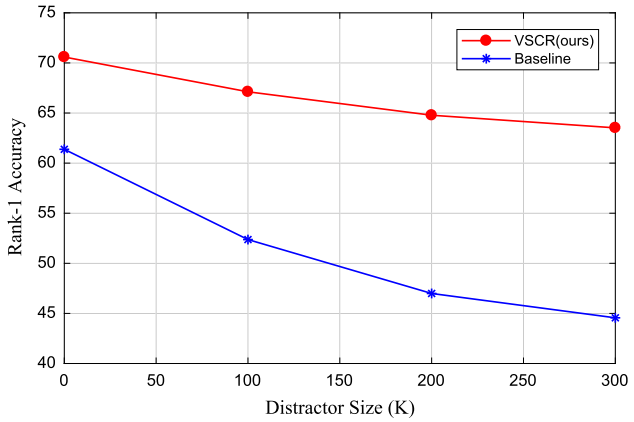


Fig. 8 Rank-1 accuracy of our method (*VSCR*) and baseline on UAV-VeID by adding different numbers of distractors

algorithms with the code provided by their authors. It can be observed that, our method achieves rank-1 accuracy of 70.59%, significantly outperforming existing ReID methods. It is interesting to observe that, existing vehicle ReID method do not perform well on UAV-VeID. Both RAM (Liu et al. 2018) and SCAN (Teng et al. 2018) extract additional regional features but perform poorly. This could be because they are designed for vehicle ReID tasks on traditional surveillance videos and do not consider the viewpoint variety and scale misalignment issues in UAV-VeID. CN-Nets (Yao et al. 2017) is designed for fine-grained instance retrieval and achieves rank-1 accuracy of 55.91%, which is the best among competitors in Table 5. The CMC curves are shown in Fig. 7.

We also experiment on the large distractor set. ReID performances achieved with different numbers of distractors are presented in Fig. 8. It is clear that, as more distractors are added to the gallery set, the ReID accuracy drops. It is interesting to see that, the performance drop of our methods is slower than the baseline. This could be because our feature is more robust to noises introduced by distracters. Figure 8 shows that, it is more challenging to perform UAV vehicle ReID on the large-scale data.

5.4.2 VeRi-776

Table 6 summarizes comparisons with recent approaches on VeRi-776. Among those compared methods RAM (Liu et al. 2018), SCAN (Teng et al. 2018), VAMI (Zhou and Shao 2018), FDA-Net (Lou et al. 2019), PRM(256 × 256) (He et al. 2019a), and AAVER (Khorramshahi et al. 2019) consider the local-part cues and design models to learn local part sensitive features. VAMI (Zhou and Shao 2018), AAVER (Khorramshahi et al. 2019), and VANet (Chu et al. 2019) consider viewpoint variation during the ReID model training. VAMI (Zhou and Shao 2018) adopts a viewpoint-aware attention model to obtain multi-view feature representation. AAVER (Khorramshahi et al. 2019) designs a viewpoint conditioned part appearance path to capture localized discriminative features on the corresponding viewpoint. VANet (Chu et al. 2019) designs a two-branch network to project a single input image into two feature spaces to enhance the feature robustness to viewpoint variances. It can be observed that, our method outperforms all of the competitors at rank-1/mAP. For example, our method achieves rank-1 accuracy of 94.11%, significantly better than the 84.27% of FDA-Net (Lou et al. 2019), 89.78% of VANet (Chu et al. 2019) and 88.97% of AAVER (Khorramshahi et al. 2019) on VeRi-776 dataset. Our method does not involve part feature extraction or keypoint detection, thus could also be easier to implement.

5.4.3 VehicleID

Table 7 summarizes comparisons with recent works on VehicleID. It is clear that, our method also shows competitive performance. PRM(256 × 256) (He et al. 2019a) integrates part and global cues and achieves similar performance to ours. It uses input size 256 × 256 (larger than our 224 × 224), which is helpful for achieving better performance.

5.4.4 VERI-Wild

Table 8 summarizes the results on VERI-Wild dataset. It is clear that ResNet50 is a strong baseline, which already achieves competitive performance. Although we use a strong baseline, our method still substantially boosts the performance. For example, it boosts the rank-1 accuracy from 78.02% to 86.29% on the large testing set of VERI-Wild, which is significantly higher than the performance of previous methods. Tables 6, 7 and 8 show that, our method also

Table 6 Comparison with recent works on VeRi-776

Methods	r=1	r=5	mAP	Backbone
GoogLeNet (Yang et al. 2015)	52.12	66.79	17.81	GoogLeNet
FACT (Liu et al. 2016d)	50.95	73.48	18.49	GoogLeNet
FACT+STR (Liu et al. 2016d)	61.44	78.78	27.77	GoogLeNet
VGG+C+T (Zhang et al. 2017)	86.41	92.91	58.78	VGG_CNN_M
OIFE (Wang et al. 2017)	65.9	87.7	48.00	GoogLeNet†
OIFE+ST (Wang et al. 2017)	68.3	89.7	51.42	GoogLeNet†
Path-LSTM (Shen et al. 2017)	83.49	90.04	58.27	ResNet50
RAM (Liu et al. 2018)	88.6	94.0	61.5	VGG_CNN_M
SCAN (Teng et al. 2018)	82.24	90.76	49.87	VGG16
VAMI (Zhou and Shao 2018)	77.03	90.82	50.13	Self-design
VAMI+ST (Zhou and Shao 2018)	85.92	91.84	61.32	Self-design
MSVR (Kanac and Zhu 2018)	88.56	–	49.30	MobileNets
PVSS (Liu et al. 2019a)	90.58	97.14	62.62	ResNet50
FDA-Net (Lou et al. 2019)	84.27	92.43	55.49	VGG_CNN_M
PRM(256 × 256) (He et al. 2019a)	92.2	**97.9**	70.2	ResNet50
VANet (Chu et al. 2019)	89.78	95.99	66.34	ResNet50
AAVER (Khorramshahi et al. 2019)	88.97	94.70	61.18	ResNet50
VSCR (ours)	**94.11**	97.85	**75.53**	ResNet50

Bold denotes the best performance

Table 7 Comparison with recent works on VehicleID

Methods	r=1	r=5	mAP	Backbone
GoogLeNet (Yang et al. 2015)	38.27	59.39	40.39	GoogLeNet
Mixed Diff+CCL (Liu et al. 2016a)	38.2	61.6	45.5	VGG_CNN_M
VGG+C+T (Zhang et al. 2017)	61.0	77.5	–	VGG_CNN_M
CLVR (Kanacı et al. 2017)	50.6	68.00	–	Inception-V3
OIFE+ST (Wang et al. 2017)	67.0	82.9	–	GoogLeNet†
RAM (Liu et al. 2018)	67.7	84.5	–	VGG_CNN_M
C2F (Guo et al. 2018)	51.4	72.2	53.0	GoogleNet
SCAN (Teng et al. 2018)	65.44	78.47	–	VGG16
VAMI (Zhou and Shao 2018)	47.34	70.29	–	Self-design
MSVR (Kanac and Zhu 2018)	63.02	73.05	–	MobileNets
FDA-Net (Lou et al. 2019)	55.53	74.65	61.84	VGG_CNN_M
PRM(256 × 256) (He et al. 2019a)	74.2	86.4	–	ResNet50
AAVER (Khorramshahi et al. 2019)	60.23	84.85	–	ResNet50
VSCR (ours)	**74.58**	**87.12**	**78.78**	ResNet50

Bold denotes the best performance

achieves promising performance on existing vehicle ReID datasets.

5.5 Visualization

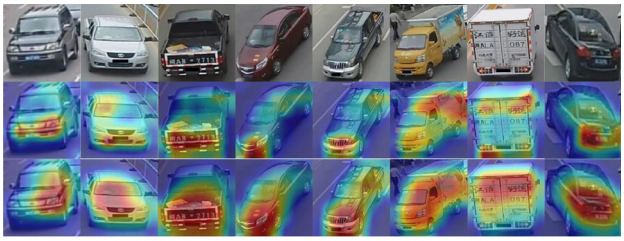
We further shows some visualizations to demonstrate the validity of our methods. Figure 9 shows the response of res5c feature maps in the ResNet50 backbone, which indicates the focused regions by the learned neural network. Response maps of baseline and our VSCR (the first branch) are compared in the second and third rows, respectively. It could be observed that, the activated regions of our VSCR contain more discriminative details, than the ones of the baseline model. This indicates that, our training strategies, i.e., the scale invariance training and viewpoint adversarial training, are effective in capturing discriminative cues from vehicle image.

Figure 10 visualizes image feature distribution of 20 vehicles randomly sampled from the UAV-VeID test set. Features extracted by baseline and our VSCR are compared. It is clear

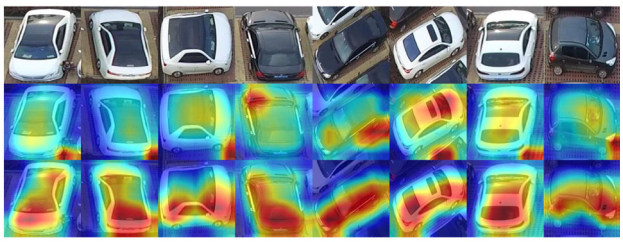
Table 8 Comparison with recent works on VERI-Wild

Settings / Methods	Small			Medium			Large			Backbone
	r=1	r=5	mAP	r=1	r=5	mAP	r=1	r=5	mAP	
GoogLeNet (Yang et al. 2015)	57.16	75.13	24.27	53.16	71.1	24.15	44.61	63.55	21.53	GoogLeNet
Triplet (Schroff et al. 2015)	44.67	63.33	15.69	40.34	58.98	13.34	33.46	51.36	9.93	GoogLeNet
Softmax (Liu et al. 2016d)	53.4	75.03	26.41	46.16	69.88	22.66	37.94	59.89	17.62	GoogLeNet
CCL (Liu et al. 2016a)	56.96	75.0	22.50	51.92	70.98	19.28	44.6	60.95	14.81	VGG_CNN_M
HDC (Yuan et al. 2017)	57.1	78.93	29.14	49.64	72.28	24.76	43.97	64.89	18.30	GoogLeNet
GSTE (Bai et al. 2018)	60.46	80.13	31.42	52.12	74.92	26.18	45.36	66.5	19.50	VGG_CNN_M
Unlabled GAN (Zhu et al. 2017)	58.06	79.6	29.86	51.58	74.42	24.71	43.63	65.52	18.23	Self-design
FDA-Net (Lou et al. 2019)	64.03	82.8	35.11	57.82	78.34	29.80	49.43	70.48	22.78	VGG_CNN_M
Baseline (ours)	88.32	92.08	67.66	82.86	90.34	63.16	78.02	87.29	56.56	ResNet50
VSCR (ours)	**93.13**	**97.70**	**75.79**	**89.68**	**96.56**	**70.47**	**86.29**	**94.60**	**64.19**	ResNet50

Bold denotes the best performance

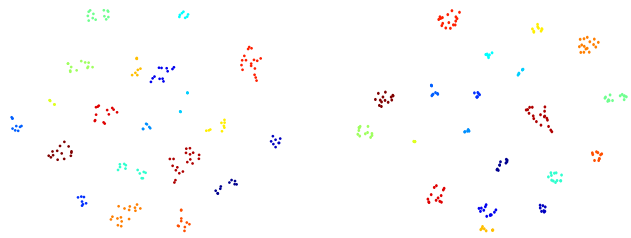


(a) VeRi-776 dataset



(b) UAV-VeID dataset

Fig. 9 Responses of feature maps generated by Grad-CAM (Ramprasaath et al. 2017). The second and third row shows responses of feature maps generated by baseline and our method, respectively. As shown in the examples, our method captures more discriminative cues



(a) Baseline **(b)** VSCR

Fig. 10 Visualization with t-SNE for feature distribution of 20 vehicles (best viewed in color). **a**, **b** Visualize the distribution of features extracted by the baseline and our VSCR, respectively. 20 vehicles are randomly sampled from the UAV-VeID testset

Fig. 11 Sample vehicle ReID results on UAV-VeID. For each example, top-10 returned results of baseline and our VSCR are shown in the first and second row, respectively. Blue boxes are query vehicles, red and green boxes denote false positives and true positives, respectively. For each query, there is only one true positive in the gallery set (Color figure online)

that, VSCR features of the same vehicle are closer to each other than the baseline feature, indicating the better robustness to variances of viewpoints and scales.

We demonstrate some vehicle ReID results on UAV-VeID and VeRi-776 in Figs. 11 and 12, respectively, where the baseline and the VSCR are compared. From Fig. 11, we can see that there exist lots of similar vehicles for each query in the gallery set. Meanwhile, the true positive exhibit different

Fig. 12 Sample vehicle ReID results on VeRi-776. For each example, top-10 returned results of baseline and our VSCR are shown in the first and second row, respectively. Blue boxes are query vehicles. Red and green boxes denote false positives and true positives, respectively (Color figure online)

viewpoints to the query. The baseline method is not effective in distinguishing those false positives. Compared with the baseline method, our VSCR method is more discriminative in identifying the same vehicle. It demonstrates that, our method learns more discriminative and robust vehicle features. Similar conclusions could be drawn from visualized results in Fig. 12.

5.6 Discussion

At present, the research on object ReID is mainly focused on person and vehicle ReID in fixed surveillance cameras. Compared with fixed cameras, cameras on UAVs are more active and flexible, thus are more suited for smart traffic surveillance. UAV person ReID is another interesting and valuable task. It faces similar challenges with UAV vehicle ReID. For instance, the same person would show substantially different viewpoints and scales in cameras of different UAVs. Therefore, UAV person ReID model also needs high robustness to viewpoint and scale changes. This work effectively alleviates viewpoint and scale variances in UAV vehicle images. Our methods thus could also be applied to other ReID tasks in UAV videos like person ReID. In our future work we will collect a large scale UAV person image dataset to verify the effectiveness of our method.

Different from existing vehicle ReID datasets taken by surveillance cameras, UAV-VeID is taken by UAVs from a bird-eye view, where the license plates are mostly invisible. This leads to difficulty in manually annotating vehicles at different times and locations. To make the dataset collection possible, as well as to simulate the real scenario as much as possible, we adopted two strategies to take UAV videos: (1) record moving vehicles at different locations and lighting conditions, and (2) record vehicles at parking lots from different viewpoints and heights. Those two strategies as well as different shooting locations and times as shown in Fig. 2 simulate considerable variances in real scenario, meanwhile make the data annotation feasible. However, compared with the real data in UAV scenario, UAV-VeID still shows weaknesses like similar backgrounds and illuminations for image of the same vehicle. More efforts are still needed to construct realistic datasets, e.g., shoot and annotate the same vehicle at different locations and times. One possible strategy is to record vehicles at different locations on a road by multiple UAVs, where extra spatial and temporal relationships can be recorded to assist the data annotation. Meanwhile, unsupervised training algorithms (e.g., Wang et al. 2019; Wang and Zhang 2020) can be developed to assist data annotation. These will be considered in our future work.

6 Conclusion

This work contributes a novel UAV-VeID dataset, which defines a more challenging and realistic vehicle ReID task in UAV videos. The UAV-VeID is expected to facilitate the development and evaluation of the vehicle ReID methods in the wild. To alleviate the negative effects of viewpoint and scale variations in vehicle images, we propose a view adversarial training strategy and a scale invariance training method to promote the robustness and discriminative power of learned deep features. Extensive experiments on UAV-VeID and existing vehicle ReID datasets show that, our approach achieves competitive performance compared with recent works.

Acknowledgements This work is supported in part by National Natural Science Foundation of China under Grant Nos. 61620106009, U20B2052, 61936011, 61931008 and 61836002, in part by the Italy-China collaboration project TALENT 2018YFE0118400, in part by Beijing Natural Science Foundation under Grant No. JQ18012, in part by Key Research Program of Frontier Sciences, CAS: QYZDJ-SSW-SYS013.

References

Avola, D., Cinque, L., Foresti, G. L., Martinel, N., Pannone, D., & Piciarelli, C. (2018). A UAV video dataset for mosaicking and change detection from low-altitude flights. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *50*, 2139–2149.

Bai, Y., Lou, Y., Gao, F., Wang, S., Wu, Y., & Duan, L. Y. (2018). Group-sensitive triplet embedding for vehicle reidentification. *TMM*, *20*(2385), 2399.

Chang, X., Hospedales, T. M., & Xiang, T. (2018). Multi-level factorisation net for person re-identification. In *CVPR*.

Chatfield, K., Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Return of the devil in the details: Delving deep into convolutional nets.

Chen, L. C., Yang, Y., Wang, J., Xu, W., & Yuille, A. L. (2016). Attention to scale: Scale-aware semantic image segmentation. In *CVPR*.

Chu, R., Sun, Y., Li, Y., Liu, Z., Zhang, C., & Wei, Y. (2019). Vehicle re-identification with viewpoint-aware metric learning. In *ICCV*.

Du, D., Qi, Y., Yu, H., Yang, Y., Duan, K., Li, G., et al. (2018). The unmanned aerial vehicle benchmark: Object detection and tracking. In *ECCV*.

Ganin, Y., & Lempitsky, V. (2014). Unsupervised domain adaptation by backpropagation.

Ganin, Y., & Lempitsky, V. (2015). Unsupervised domain adaptation by backpropagation. In *ICML*.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., et al. (2016). Domain-adversarial training of neural networks. *JMLR*.

Girisha, S., Pai, M. M., Verma, U., & Pai, R. M. (2019). Performance analysis of semantic segmentation algorithms for finely annotated new uav aerial video dataset (manipaluavid). *IEEE Access*, *7*, 136239–136253.

Guo, H., Zhao, C., Liu, Z., Wang, J., Lu, H. (2018). Learning coarse-to-fine structured feature embedding for vehicle re-identification. In *AAAI*.

He, J., Deng, Z., & Qiao, Y. (2019b). Dynamic multi-scale filters for semantic segmentation. In *ICCV*.

He, B., Li, J., Zhao, Y., & Tian, Y. (2019a). Part-regularized near-duplicate vehicle re-identification. In *CVPR*.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *CVPR*.

Hsieh, M. R., Lin, Y. L., & Hsu, W. H. (2017). Drone-based object counting by spatially regularized regional proposal network. In *ICCV*.

Huang, G., & Chen, D. (2018). Multi-scale dense networks for resource efficient image classification. In *ICLR*.

Kanac, A., & Zhu, X. (2018). Vehicle re-identification in context. In *GCPR*.

Kanacı, A., Zhu, X., & Gong, S. (2017). Vehicle reidentification by fine-grained cross-level deep learning. In *BMVC*.

Khorramshahi, P., Kumar, A., Peri, N., Rambhatla, S. S., Chen, J. C., & Chellappa, R. (2019). A dual path modelwith adaptive attention for vehicle re-identification. In *ICCV*.

Li, Y., Chen, Y., Wang, N., & Zhang, Z. (2019). Scale-aware trident networks for object detection. In *ICCV*.

Li, W., Zhu, X., & Gong, S. (2018). Harmonious attention network for person re-identification. In *CVPR*.

Liu, X., Liu, W., Ma, H., & Fu, H. (2016c). Large-scale vehicle re-identification in urban surveillance videos. In *ICME*.

Liu, X., Liu, W., Ma, H., & Li, S. (2019b). PVSS: A progressive vehicle search system for video surveillance networks.

Liu, X., Liu, W., Mei, T., & Ma, H. (2016d). A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In *ECCV*.

Liu, L., Qiu, Z., Li, G., Liu, S., Ouyang, W., & Lin, L. (2019a). Crowd counting with deep structured scale integration network. In *ICCV*.

Liu, H., Tian, Y., Yang, Y., Pang, L., & Huang, T. (2016a). Deep relative distance learning: Tell the difference between similar vehicles. In *CVPR*.

Liu, W., Wen, Y., Yu, Z., & Yang, M. (2016b). Large-margin softmax loss for convolutional neural networks. In *ICML*.

Liu, X., Zhang, S., Huang, Q., & Gao, W. (2018). RAM: A region-aware deep model for vehicle re-identification. In *ICME*.

Long, M., Cao, Z., Wang, J., & Jordan, M. I. (2018). Conditional adversarial domain adaptation. In *NIPS*.

Lou, Y., Bai, Y., Liu, J., Wang, S., & Duan, L. (2019). Veri-wild: A large dataset and a new method for vehicle re-identification in the wild. In *CVPR*.

Lu, J., Yang, J., Batra, D., & Parikh, D. (2016). Hierarchical question-image co-attention for visual question answering. In *NIPS*.

Mueller, M., Smith, N., & Ghanem, B. (2016). A benchmark and simulator for UAV tracking. In *ECCV*.

Pei, Z., Cao, Z., Long, M., & Wang, J. (2018). Multi-adversarial domain adaptation. In *AAAI*.

Qian, X., Fu, Y., Jiang, Y. G., Xiang, T., & Xue, X. (2017). Multi-scale deep learning architectures for person re-identification. In *ICCV*.

Ramprasaath, R. S., Michael, C., Abhishek, D., Ramakrishna, V., Devi, P., & Dhruv, B. (2017). Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *ICCV*.

Redmon, J., & Farhadi, A. (2017). Yolo9000: Better, faster, stronger. In *CVPR*.

Robicquet, A., Sadeghian, A., Alahi, A., & Savarese, S. (2016). Learning social etiquette: Human trajectory understanding in crowded scenes. *In* ECCV.

Schroff, F., Kalenichenko, D., & Philbin, J. (2015) FaceNet: A unified embedding for face recognition and clustering. In *CVPR*.

Shen, Y., Xiao, T., Li, H., Yi, S., & Wang, X. (2017). Learning deep neural networks for vehicle re-id with visual-spatio-temporal path proposals. In *ICCV*.

Sun, Y., Zheng, L., Yang, Y., Tian, Q., & Wang, S. (2018). Beyond part models person retrieval with refined part pooling. In *ECCV*.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., et al. (2015). Going deeper with convolutions. In *CVPR*.

Tan, W., Yan, B., & Bare, B. (2018). Feature super-resolution: Make machine see more clearly. In *CVPR*.

Teng, S., Liu, X., Zhang, S., & Huang, Q. (2018) SCAN: Spatial and channel attention network for vehicle re-identification. In *PCM*.

Teng, S., Zhang, S., Huang, Q., & Sebe, N. (2020). Multi-view spatial attention embedding for vehicle re-identification. TCSVT.

Tzeng, E., Hoffman, J., Saenko, K., & Darrell, T. (2017). Adversarial discriminative domain adaptation. In *CVPR*.

Wang, D., & Zhang, S. (2020). Unsupervised person re-identification via multi-label classification. In *CVPR*.

Wang, X., Jabri, A., & Efros, A. A. (2019). Learning correspondence from the cycle-consistency of time. In *CVPR*.

Wang, Z., Tang, L., Liu, X., Yao, Z., Yi, S., Shao, J., et al. (2017). Orientation invariant feature embedding and spatial temporal regularization for vehicle re-identification. In *CVPR*.

Wei, L., Zhang, S., Gao, W., & Tian, Q. (2017a). Person transfer gan to bridge domain gap for person re-identification. In *CVPR*.

Wei, L., Zhang, S., Yao, H., Gao, W., & Tian, Q. (2017b). Glad: Global-local-alignment descriptor for pedestrian retrieval. In *ACM MM*.

Wen, Y., Zhang, K., Li, Z., & Qiao, Y. (2016). A discriminative feature learning approach for deep face recognition. In *ECCV*.

Yan, K., Tian, Y., Wang, Y., Zeng, W., & Huang, T. (2017). Exploiting multi-grain ranking constraints for precisely searching visually-similar vehicles. In *ICCV*.

Yang, L., Luo, P., Change Loy, C., & Tang, X. (2015). A large-scale car dataset for fine-grained categorization and verification. In *CVPR*.

Yao, H., Zhang, S., Zhang, Y., Li, J., & Tian, Q. (2017). One-shot fine-grained instance retrieval. In *ACM MM*.

Yuan, Y., Yang, K., & Zhang, C. (2017). Hard-aware deeply cascaded embedding. In *ICCV*.

Zhang, Y., Liu, D., & Zha, Z. J. (2017). Improving triplet-wise training of convolutional neural network for vehicle re-identification. In *ICME*.

Zhou, Y., & Shao, L. (2018). Aware attentive multi-view inference for vehicle re-identification. In *CVPR*.
Zhou, K., Yang, Y., Cavallaro, A., & Xiang, T. (2019). Omni-scale feature learning for person re-identification. In *ICCV*.
Zhu, J. Y., Taesung, P., Phillip, I., & Alexei, A. E. (2017). Unpaired imageto-image translation using cycle-consistent adversarial networks. In *ICCV*.
Zhu, P., Wen, L., Bian, X., Ling, H., & Hu, Q. (2018a). Vision meets drones: A challenge.
Zhu, P., Wen, L., Du, D., Bian, X., Hu, Q., & Ling, H. (2020). Vision meets drones: Past, present and future.
Zhu, Z., Wu, W., Zou, W., & Yan, J. (2018b). End-to-end flow correlation tracking with spatial-temporal attention. In *CVPR*.