ISPRS Journal of Photogrammetry and Remote Sensing 180 (2021) 283–295

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

Multi-scale adversarial network for vehicle detection in UAV imagery

Ruiqian Zhang [a,e], Shawn Newsam [b], Zhenfeng Shao [c,*], Xiao Huang [d], Jiaming Wang [c], Deren Li [a,c]

[a] School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, 430079, China
[b] Electrical Engineering and Computer Science, University of California, Merced, 95343, USA
[c] State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, 430079, China
[d] Department of Geosciences, University of Arkansas, Fayetteville, AR, 72701, USA
[e] Chinese Academy of Surveying and Mapping, Beijing, 100036, China

ARTICLE INFO

Keywords:
Vehicle detection
UAV imagery
Multi-scale structure
Adversarial network
Domain adaptation

ABSTRACT

Vehicle detection in Unmanned Aerial Vehicle (UAV) imagery plays a crucial role in a variety of applications. However, UAVs are usually small, very maneuverable, and can take images from a variety of viewpoints and heights, leading to large differences in vehicle appearance and size. To address the vehicle detection challenge with such diversity in UAV images, we seek to align features between different viewpoints, illumination, weather, and background using remote sensing imagery as an anchor. Following this domain adaptation concept, we propose a multi-scale adversarial network, consisting of a deep convolutional feature extractor, a multi-scale discriminator, and a vehicle detection network. Specifically, the feature extractor is a Siamese network with one path for the UAV imagery and another for the satellite imagery. The shared weights in this sub-network allow us to exploit the large collections of labeled remote sensing imagery for improved vehicle detection in UAV imagery. Experimental results suggest that our proposed algorithm improves the vehicle detection accuracy in the UAVDT dataset and VisDrone dataset. The proposed model achieves great performance in images taken from different perspectives, at different altitudes, and under different imaging situations.

1. Introduction

Unmanned Aerial Vehicles (UAVs) are a new and prominent remote sensing platform (Colomina and Molina, 2014; Zhang et al., 2017b). Made possible by advancements in battery and manufacturing technology, UAVs are demonstrating their potential for a broad range of societally important applications such as precision agriculture (Gevaert et al., 2015), environmental monitoring (B et al., 2020), disaster warning and recovery (Li et al., 2012), and wildlife protection (Yang et al., 2014). Key to many of these applications is being able to automatically detect, classify, and track objects in the UAV imagery, supported by the development of new techniques that involve computer vision and deep learning.

In this paper, we focus on the problem of vehicle detection in UAV imagery; that is, automatically identifying and localizing objects of interest. While there has been significant progress in standard imagery, especially using deep learning, vehicle detection in UAV imagery remains difficult due to the large variation in object size, viewpoint, environmental conditions including illumination and weather, and background. Most object detection approaches are designed for ground-level images taken from a horizontal viewpoint, such as the images in the ImageNet dataset (Krizhevsky et al., 2012). However, UAVs are small, very maneuverable (Zhou et al., 2009; Kalantar et al., 2017; Benjamin et al., 2018), and can take images from a variety of viewpoints and heights, leading to large differences in vehicle appearance and size (like objects in Fig. 1). The diversity of applications also results in complex and varying backgrounds. Finally, UAVs are usually flown outside where weather and the wide range of ambient illumination (e.g., night versus day) can greatly affect the quality of the images. This variation presents a significant challenge in particular to supervised learning based approaches that rely on training examples that are representative of all possible scenarios. The large amounts of training data required by deep learning algorithms further exacerbate the situation.

A standard approach to dealing with variation in appearance is to perform alignment or normalization by generative adversarial networks to the images before they are input to the object detection module. The original generative adversarial networks are proposed for generating random samples, which is widely used in image stylization (Zhu et al., 2017), image synthesis (Zhang et al., 2017a) and super resolution (Ledig et al., 2017) tasks. However, for some tasks, the generation of random samples is not suitable. Taigman et al. (2017) design a new adversarial network as a general tool to measure the equivalence

* Corresponding author.
E-mail address: shaozhenfeng@whu.edu.cn (Z. Shao).

https://doi.org/10.1016/j.isprsjprs.2021.08.002
Received 26 November 2020; Received in revised form 21 June 2021; Accepted 2 August 2021
Available online 1 September 2021

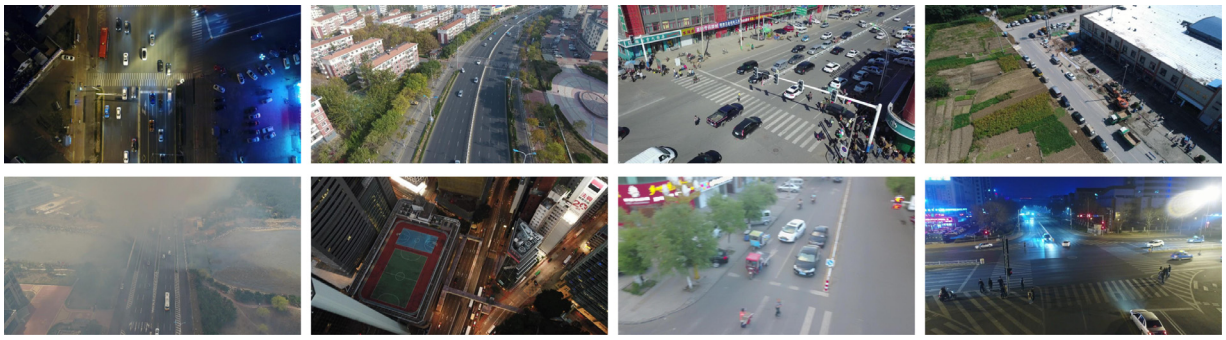

Fig. 1. Examples of UAV images in UAVDT (Du et al., 2018) and VisDrone (Zhu et al., 2018) datasets. The first line shows images with large differences in vehicle appearance and size, and second line presents images in complex and varying backgrounds, weather and illumination.

between distributions. Ajakan et al. (2014) link the adversarial loss to the H-divergence between two distributions and achieved unsupervised domain adaptation with an adversarial network. Motivated by theory on domain adaptation (Bousmalis et al., 2016) where a transferable feature is one for which an algorithm cannot learn to identify the domain of origin of the input observation, Hung et al. (2018) integrate adversarial feature learning into semi-supervised semantic segmentation and prove the utility of feature learning with an adversarial network for segmentation tasks. However, working in the two dimensional image space limits capability of such transformations that are possible to spatial (e.g., affine) or intensity mappings. In addition, learning such mapping remains to be a challenging task in a supervised framework.

We instead take a different approach to address the large variation in appearance in UAV imagery for improved vehicle detection. We present the following key insights. First, the current state-of-the-art vehicle detection pipelines generally consist of two components: a feature extraction network and a vehicle detection network. The performance of the vehicle detection network could be greatly improved if it was isolated from the variation in appearance. This motivates us to perform the alignment in the feature space instead of the image space, which allows for a richer set of transformations that can be learned in a supervised manner. Second, while there might be a large variation in appearance in UAV imagery, there is much less in traditional remote sensing imagery, such as the ones taken from satellite or aircraft. This imagery can thus serve as a common anchor or reference for aligning the UAV features. The rich training remote sensing images further contribute to the robustness of such anchor training.

Following the above rationale, we propose a novel multi-scale adversarial network for improved vehicle detection in UAV imagery. Our network performs feature alignment through adversarial learning similar to how Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) have been used to align features between different image domains (e.g., photographs and paintings) by having two sub-networks play a min–max game in the training process. We, however, seek to align features between different viewpoints, illumination, weather, and background using remote sensing imagery as an anchor. Our network follows a multi-scale design, allowing it to deal with the variation in object size.

The proposed framework consists of three sub-networks: a deep convolutional feature extractor, a multi-scale discriminator, and a vehicle detection network. The feature extractor is a Siamese network with one path for the UAV imagery and another for the remote sensing imagery. The shared weights in this sub-network allow us to exploit the large collections of labeled remote sensing imagery for improved vehicle detection in UAV imagery. The multi-scale discriminator performs feature alignment by forcing the extractor, in an adversarial manner, to learn features that are indistinguishable as to whether they are from UAV or remote sensing imagery. This results in feature alignment not only between the UAV and remote sensing imagery but also between the UAV imagery itself. Aligning the UAV features to the reference remote sensing features, which do not exhibit variation due to viewpoint, illumination, etc., results in UAV features that are aligned to themselves and thus more uniform. This greatly improves the performance of the vehicle detection network which takes the features as major inputs. All three sub-networks follow a multi-scale design, benefiting the capability of the model in explicitly handling variation in object size.

We demonstrate the effectiveness of our proposed multi-scale adversarial network on multiple UAV vehicle detection benchmark datasets. In particular, we show that jointly training with the UAV and remote sensing datasets results in improved performance over standard sequential training.

Our key contributions are summarized as follows:

- We design a two-path multi-scale feature extraction network, allowing us to exploit large collections of labeled remote sensing imagery with scale diversity when training object detectors for UAV imagery.
- We develop a novel adversarial framework that uses a discriminator to align features between UAV and remote sensing imagery. This subsequently results in improved feature alignment between the UAV images themselves, such as images taken from different viewpoints, under different environmental conditions, making the vehicle detection more robust to variation in appearance in the UAV imagery.
- We demonstrate the effectiveness of our approach on the UAVDT (Du et al., 2018) and VisDrone (Zhu et al., 2018) datasets.

2. Related work

2.1. Vehicle detection for natural images

Vehicle detection is generally carried out as a category of object detection for natural images, where a fair amount of deep learning based research has been conducted in recent years. Inspired by the recent successes of deep convolutional neural networks (CNNs) for image classification (Krizhevsky et al., 2012), Girshick et al. (2015) propose an Regions with CNNs (RCNN) detector that adopts a selective searching approach to obtain proposals, uses CNNs as feature extractors, and applies a Support Vector Machine (SVM) classifier and a bounding box regression network. Different from other machine learning methods, RCNN does not need hand-crafted features for each object category. Instead, the CNNs learn to extract appropriate features from a variety of objects during training. Despite that the performance of RCNN surpasses previous machine learning methods, its speed is limited by the redundant computations for overlapping proposals. Further, Fast RCNN (Girshick, 2015) and Faster RCNN (Ren et al., 2015) are proposed to increase the speed of the detector by combining the region proposal network and the feature extraction network into one single network. Faster R-CNN can detect objects with high speed and still

achieve state-of-the-art performance on different detection challenges. Given its success, many algorithms have followed the design Faster RCNN, such as R-FCN (Dai et al., 2016) and Mask R-CNN (He et al., 2017).

Theoretically, the aforementioned methods belong to two-step networks, as they all use region proposals. There are other methods that detect objects with a one-step network. Compared with region proposal based algorithms, these methods are regression/classification based, including MultiBox (Liu et al., 2016), AttentionNet (Yoo et al., 2015), G-CNN (Najibi et al., 2016), YOLO (Redmon et al., 2016), SSD (Liu et al., 2016), YOLOv2 (Redmon and Farhadi, 2017), YOLOv3 (Redmon and Farhadi, 2018), and DSOD (Shen et al., 2017). One of the most popular methods, YOLO (Redmon et al., 2016), predicts bounding boxes and class probabilities directly from full images in one-step. Since the entire detection pipeline is a single network, it can be directly optimized in an end-to-end manner. YOLO is considerably fast with the capability of processing images in real-time at a speed of 45 frames per second (Redmon et al., 2016). Compared to the two-step detection systems, YOLO makes more localization errors but is less likely to predict false positives in the background. Following the success of YOLO, YOLOv2 and YOLOv3 were progressively developed to improve the original version of YOLO. With a different network design and multi-scale training, YOLOv3 outperforms state-of-the-art methods (e.g., Faster RCNN) with a considerably faster speed.

Recently, a considerable number of algorithms have been proposed to detect objects in natural images. Pang et al. (2019) introduce an overall balanced design into sampling, feature extracting, and objective level in object detection network. Wang et al. (2020) develop a Side-Aware Boundary Localization approach for improving the precision of bounding box regression. Scholars in Tian et al. (2019) propose a fully convolutional one-stage object detector (FCOS) without the predefined set of anchor boxes to avoid the complicated computation related to anchor boxes. Tan et al. (2020) adopt better backbones and a compound scaling method to uniformly scale the resolution, depth, and width at the same time.

In UAV scenarios, fast detection is preferred. Thus, we utilize a single neural network based on YOLOv3 (Redmon and Farhadi, 2018) as our baseline. We enhance the YOLOv3 backbone with a multi-scale discriminator, aiming to align features with multi-scale objects and incorporate satellite imagery during training to align features between various viewpoints, illuminating conditions, weather, background, etc.

2.2. Vehicle detection in aerial imagery

Aerial images tend to have an overhead perspective and large fields of view due to their bird's eye view of the target from high altitudes (Xia et al., 2018; Audebert et al., 2018). To detect objects from aerial imagery, many approaches incorporate frame and background information into the detection task. Rodríguez-Canosa et al. (2012) develop a method that combines camera motion estimation based on static point features with optical flow comparison to determine the pixels that belong to dynamic objects. Aslani and Mahdavi-Nasab (2013) present a method with optical flow and median filters to detect moving objects in aerial imagery. Jabar et al. (2015) and Kamate and Yilmazer (2015) detect moving objects using frame differences in regions with near key points for optical flow field or a mean-shift tracker. However, approaches that rely on frame differences and background subtraction are only applicable when overlapping images exist. Further, these approaches fail to achieve great performance for imagery with complex backgrounds or from multiple viewpoints.

In order to solve the above problems and improve the performance on vehicle detection, many papers design networks only on vehicle object. Moranduzzo and Melgani (2014) design a method that first identifies asphalted areas and uses filtering operations in horizontal and vertical directions with a catalog of cars as a reference to detect vehicles. Zhou et al. (2018) propose a solution based on the bag-of-words (BoW) model and an orientation-aware scanning mechanism to identify vehicles following the "one-window-one-car" concept. Despite the success of these methods, they tend to consume a lot of manpower and computational resources. Over the past two decades, the rapid development of machine learning and deep learning greatly facilitates the improvement of vehicle detection from aerial imagery. More recent efforts involve the application of deep learning algorithms. Chen et al. (2014) detect cars with a deep neural network running on the graphics processor united in a sliding-window approach. Li et al. (2019) propose a rotatable region-based residual network based on rotatable region proposal network and rotatable detection network architectures. Although the integration of machine learning and deep learning approaches achieve improved vehicle detection performance, robust vehicle detection models require a large number of heterogeneous UAV images, which are often difficult to obtain.

2.3. Domain adaptation for cross-view imagery

Recent years, intensive researches have been proposed in domain adaptation field, which aims to transfer knowledge from a domain with adequate labeled samples to a domain with scarce labeled samples (Shermin et al., 2020; Al-Moslmi et al., 2017; Li et al., 2020). Due to variations in camera perspective, great discrepancies can exist among vehicles and scenes in different images. Efforts have been made to address these differences in cross-view imagery in computer vision tasks, such as action recognition, person re-identification, segmentation, and so on. Traditional works on cross-view recognition focus mostly on model reconstruction between multiple views or designing hand-crafted features to recognize the same action from multiple views. Bak et al. (2010) perform cross-view person recognition using Haar-based and DCD-based signatures. Liu et al. (2011) propose a novel bipartite-graph-based approach to learn bilingual-words from two view-dependent vocabularies in an unsupervised manner. However, deriving the reconstruction model and learning the features of different actions tend to be computationally intensive, leading to long training time. Zheng et al. (2015) propose a data-driven distance metric (DDDM) method for cross-view person re-identification, which re-exploits the training data to adjust the metric for each query-gallery pair. Chen et al. (2016) formulate an asymmetric distance model to transform the unmatched features from each view into an integrated space. The utilization of model reconstruction and hand-crafted features facilitates the integration of images in multiple views. However, it takes a long time to design models and features which limits the application of task and results in less robustness to problems with large numbers of classes.

With the advent of deep learning, domain adaptation methods using neural networks for cross-view recognition have been proposed. Peng et al. (2017) use deep neural networks to generate frontal views from a single frontal face with the combination of rich feature embedding and reconstruction metric learning to recognize faces from different views. Huang et al. (2017) designed a Two-Pathway Generative Adversarial Network (TP-GAN) for photorealistic frontal view synthesis by jointly perceiving global structures and local details, which not only presents compelling perceptual results but also outperforms state-of-the-art results on large pose face recognition. Zhang et al. (2019) apply top variational auto-encoders, cross-view alignment, and dual GANs to integrate a series of non-linear transformations into a shared latent space, allowing a comparable matching across various camera views.

However, domain adaptation for cross-view vehicle detection in UAV imagery remains under-exploited. The bird's eye views from UAVs with relatively high altitudes lead to overhead viewpoints. Compared with aerial imagery from satellites, images in UAVs are considerably more dynamic (Zhou et al., 2009; Kalantar et al., 2017; Benjamin et al., 2018). Thus, multi-view vehicle detection is a fundamental yet challenging issue in UAV imagery.

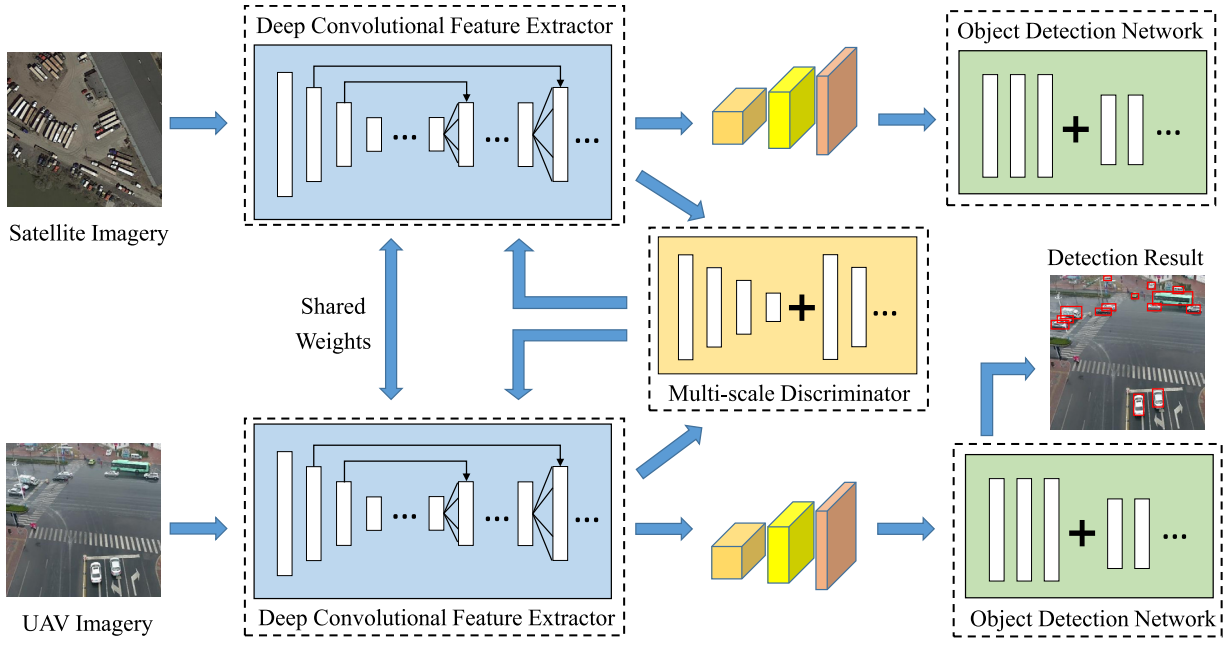

Fig. 2. The overview of the proposed multi-scale adversarial network. It consists of a deep convolutional feature extractor, a multi-scale discriminator, and a vehicle detection network. The deep convolutional feature extractor is fed with two-path imagery together, and learns to align features between satellite images and UAV images.

3. Proposed method

Vehicles in UAV imagery often have varying sizes, viewpoints, appearances, and complex backgrounds with large inter-image differences, leading to great difficulty in learning and aligning the feature distributions corresponding to different instances of the same object class. We, therefore, propose an adversarial network with the following objectives:

(1) To improve detection generalization, we introduce labeled satellite imagery into the training process of our UAV vehicle detector. The feature extractor for satellite images and the feature extractor for UAV image feature extractor share weights, leading to the adaption between the two types of images.

(2) To facilitate the learning of collaborative features beyond perspectives and altitudes, we design a discriminator following a multi-dimensional design. It can integrate three features with three dimensions to align object features with diverse sizes. The discriminator and extractor compete against each other to improve the performance during training in the designed adversarial network.

Fig. 2 illustrates the overview of the proposed multi-scale Adversarial Network, which consists of a deep convolutional feature extractor, a multi-scale discriminator, and a vehicle detection network. In the following sections, we describe the detailed information for deep convolutional feature extractor. Then present the creative design and loss function information for multi-scale discriminator. Finally, the vehicle detection network integrate architecture and method for back propagation is be described.

3.1. Deep convolutional feature extractor

The deep convolutional feature extractor is constructed following a two-path design, as shown in Fig. 2. The first path takes satellite imagery as input and outputs multiple dimensional features from satellite imagery. The second path shares the same network structure with UAV imagery as input. These two paths share weights. We adopt the deep convolutional feature extractor with the same architecture in $DarkNet-53$ (Redmon and Farhadi, 2018) as the backbone. It has 53 convolutional layers and uses successive 3×3 and 1×1 convolutional layers as well as some shortcut connections, called Residual Blocks. Note that the $DarkNet-53$ architecture in our multi-scale adversarial network can be replaced with other similar backbones like $DarkNet-19$ (Redmon and Farhadi, 2017). In order to coordinate the multi-scale discriminator, we output three different layers in the feature extractor, representing three-dimensional features. With these multiple features, our adversarial network can integrate high-dimensional and low-dimensional features into one network, taking multi-scale vehicles into consideration by extracting objects with heterogeneous sizes and shapes.

The detailed architecture of the deep convolutional feature extractor is shown in Fig. 3. The entire network structure before layer 74 is completed by convolutional layers and residual blocks, without any pooling layer. Each convolutional layer packages convolution, batch normalization, and LeakyReLU operations. The residual block packages a set of convolutional layers and shortcut operators. After layer 74, the structure is completed by convolutional layers, upsample layers, and route layers that convert from feature extraction to feature integration. The upsample layers apply bilinear upsampling by stride to expand the size of the previous feature. The route layers operate concatenation by channel dimension with two feature maps.

Adopted from ResNet (He et al., 2016), the design of a residual block is shown in Fig. 4. By using residual blocks, the inputs (dark blue block in Fig. 4) and outputs (brown block in Fig. 4) are made generally consistent. In residual blocks, the shortcut layer is designed to calculate the difference between the connected layer. To solve the issue that layers with different sizes fail to be connected easily, we propose a 1 convolutional layer, followed by a 3 convolutional layer. With this design, the output feature (shown as the brown block in Fig. 4) has the same size as the input one (shown as the dark blue block in Fig. 4). The residual block in the feature extractor not only convert the layer-by-layer training of the deep neural network into phase-by-stage training, but also divides the deep neural network into multiple sub-segments. Each of the residual blocks contains a relatively shallow network layer with shortcut connections. As the difference is calculated within every block, the general loss of the extractor is reduced, which solves the issue of gradient dispersion or gradient explosion of the network.

For the three output feature maps, the first feature map with 1024 channels (shown as the orange block, layer 74 in Fig. 3) outputs from the layer 74 feature. The height and width are all set as 13. With 1024

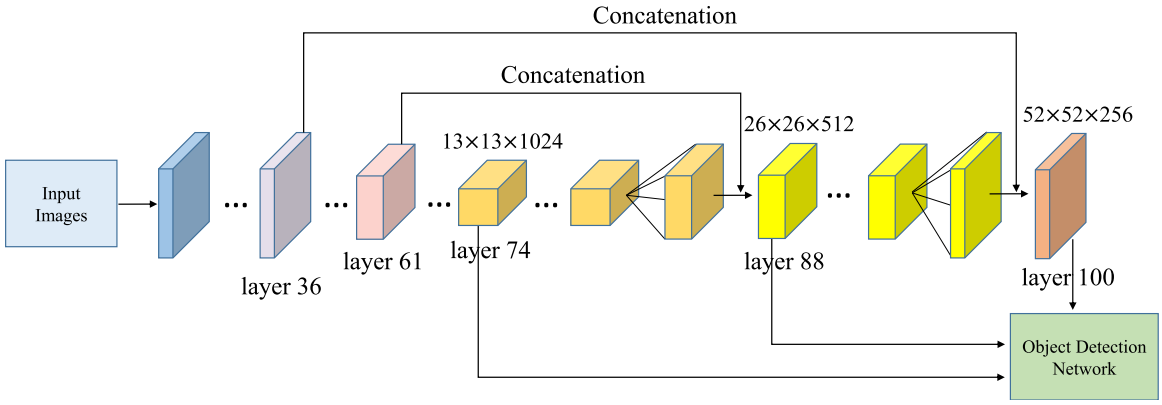

Fig. 3. The architecture of the deep convolutional feature extractor for each path. Layer 36, layer 61, and layer 74 are considered as output features of the discriminator with multi-scale. And vehicle detection network uses layer 74, layer 88, and layer 100 as inputs. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

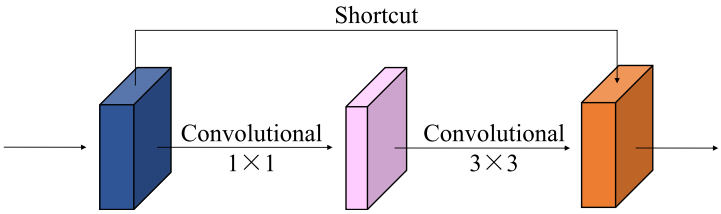

Fig. 4. The detailed structure of a residual block. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

channels, it has the ability to extract high-dimensional features but without detailed information. Given its large receptive field size, it is suitable for detecting objects with relatively large size. In order to fuse lower-dimensional features together, we adopt an upsampling layer, followed by a concatenation operation. The upsampling layer does not change the size of the channel but doubles the size of feature maps. The concatenate operation combines two features channel-wise. The output feature for concatenation has the same size as the input features, but the channel size equals the sum of two input channels. Followed by combination, a convolutional operation is applied to smooth the number of channels from the sum of two input sizes to the input size. The layer after the concatenation and convolutional operation is output as the feature maps. The deep convolutional feature extractor contains two concatenation processes. One concatenation operation fuses layer 88 and obtains the second feature maps with a size of $26 \times 26 \times 512$ (shown as the yellow block in Fig. 3). The other concatenation operation fuses layer 100 and obtains the third feature maps with a size of $52 \times 52 \times 256$ (shown as the red block in Fig. 3). In the end, the deep convolutional feature extractor outputs a total of three features with different sizes.

3.2. Multi-scale discriminator

Inspired by generative adversarial networks (GAN) (Goodfellow et al., 2014) that has been widely used in computer vision tasks, we design a multi-scale discriminator. The goal of the GAN is to learn a generative distribution $P_z(z)$ from data z that matches the real data distribution $P_{data(x)}$ from data x. Specifically, GAN can learn a generative network G and discriminative model D, where G generates samples from the generator distribution $P_z(z)$ and D learns to determine whether a sample is from $P_z(z)$ or $P_{data(x)}$. Via a training process, the discriminator D and the generator G compete against each other by playing a minimal maximization game on the value function $V(G, D)$:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim P_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim P_z(z)}[\log(1 - D(G(z)))].$$

In our case, we design a multi-scale discriminator to facilitate feature adaptation between satellite imagery and UAV imagery and a cross-view adaptation for vehicle objects of certain categories. It uses layer 36 ($52 \times 52 \times 256$), layer 61 ($26 \times 26 \times 512$) and layer 74 ($13 \times 13 \times 1024$) in deep convolutional feature extractor as inputs. The structure of the multi-scale discriminator is shown in Fig. 5. It contains 3 concatenate substructures, each of which consists of 4 convolution layers with 3×3 kernels. We set $stride = 2$ and $padding = 1$. Each convolution layer is followed by a Leaky-ReLU (Xu et al., 2015) parameterized by 0.2 and a Dropout layer parameterized by 0.25. To stabilize the training of the multi-scale discriminator, we use the spectral-normalization layer (Miyato et al., 2018) for weight normalization on each layer of the discriminator. However, no batch-normalization layer is adopted, as a study has proved that it only performs well when the batch size is sufficiently large (Hung et al., 2018).

Since the three substructures are fed with different features, we construct different channels through the layers. In the first path, the four convolutional layers are designed with $\{512, 1024, 512, 128\}$ channels, given the input feature X_1 of size $52 \times 52 \times 256$. The second one (input channel of 512) and the third one (input channel of 1024) are all presented with $\{1024, 512, 256, 128\}$ channels. After the 4 convolutional layers in each path, a linear layer is applied to perform a binary classification task for all channels.

During the training process, the multi-scale discriminator aims to discriminate whether the output feature is from satellite imagery or UAV imagery from the extractor. x_s and x_u are the representations for satellite and UAV imagery, respectively. The extractor aims to learn a function G_e with which the UAV imagery feature $G_e(x_u)$ can have a similar distribution with satellite image features $G_e(x_s)$. An adversarial network is therefore formed, consisting of the multi-scale discriminator and the deep feature extractor. They compete against each other to

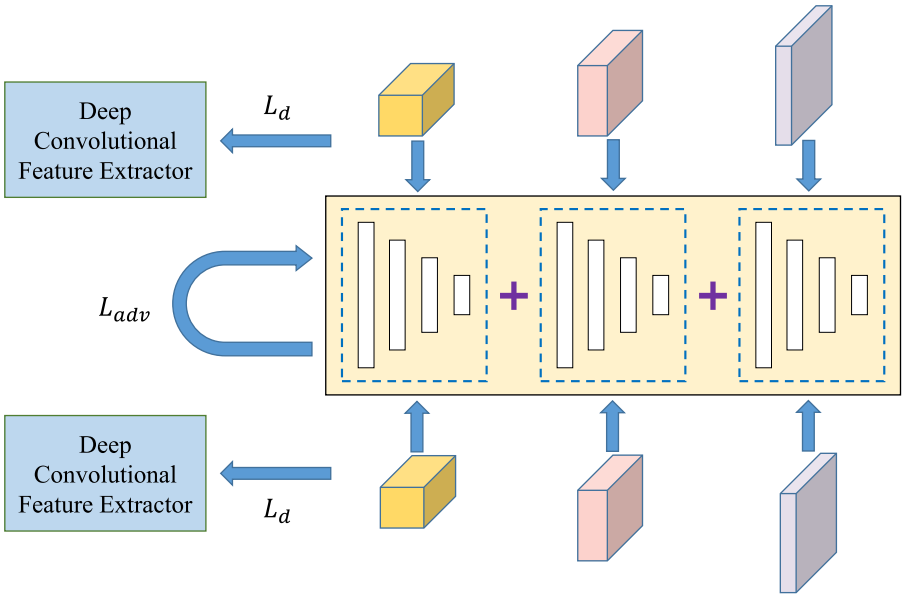

Fig. 5. The architecture of multi-scale discriminator. Multi-scale features are input into discriminator respectively.

obtain a highly performed feature extractor for vehicle detection by playing a minimal maximization game that can be formulated as :

$$\min_{G_e}\max_{D} V(D,G_e) = \mathbb{E}_{G_e(x_s)\sim P_{data}(G_e(x_s))}[\log D(G_e(x_s))] + \mathbb{E}_{G_e(x_u)\sim P_u(G_e(x_u))}[\log(1-D(G_e(x_u)))].$$

The discriminator is with $N = 3$ scales, and each scale substructure has its own cross-entropy loss L_n during training. The integration of cross-entropy loss L_d for the discriminator can be calculated via the following formulas:

$$L_n = -\log(D_n(G_e(x_s))) - \log(1 - D_n(G_e(x_u))),$$

$$L_d = \sum_{n=1}^{N} L_n.$$

Using the loss function L_d, the discriminator learns to distinguish the difference between features from satellite images and features from UAV images. We also define an adversarial loss function L_{adv} to train the deep convolutional feature extractor. With the adversarial loss, the feature extractor learns to fool the discriminator by maximizing the similarity of the satellite image features and UAV image features. The L_{adv} is formulated as:

$$L_{adv} = \sum_{n=1}^{N} \lambda_{adv}(-\log(D_n(G_e(x_u)))).$$

where λ_{adv} represents the weights for minimizing the proposed adversarial loss function in the designed multi-scale discriminator. The L_{adv} denotes the backpropagation process in the deep convolutional feature extractor.

3.3. Vehicle detection network

Vehicle detection network is also a multi-scale network. With the three-scale features inputting into the network, each feature is followed by a set of convolutional layers and a regression layer. The convolutional layers extract and smooth the channels of features and output each feature with 75 channels. Thus, after convolutional layers, features contain sizes of $\{13 \times 13 \times 75\}$, $\{26 \times 26 \times 75\}$ and $\{52 \times 52 \times 75\}$. For the regression layer, we set anchor boxes to predict initial object positions following the design by YOLOv3. Since the previous layers are all convolutional layers, the regression layers can simply predict these offsets at the pixel level in feature maps. Each pixel is treated as a grid cell, and the anchor boxes are considered as masks covering each grid cell in feature maps. Given that the features from an image are with 3 scales, the anchor boxes with a total of 9 different sizes. Each feature has 3 different anchors, and they can be set using the Kmeans method or other similar methods. The anchor boxes are regarded as masks on each position of features, and the vehicle detection network further predicts offsets and confidences for anchor boxes given the ground truth.

Specifically, we apply anchor boxes based on the experiments fine-tuned on ImageNet dataset (Krizhevsky et al., 2012): $(116 \times 90), (156 \times 198), (373 \times 326)$ on the smallest 13×13 feature map, which contains anchors with the largest receptive field and more suitable for detecting larger vehicles. On the medium 26×26 feature map, we use anchors with a medium receptive field as (30×61), (62×45), (59×119), suitable for detecting medium sized vehicles. The largest 52×52 feature map uses the smallest receptive field anchor boxes with $(10 \times 13), (16 \times 30), (33 \times 23)$ for detecting smaller vehicles. Here, the anchors are defined as the width and height corresponding to the size of the resized image, i.e., 416.

In the designed vehicle detection network, we stable the network by following the direct location prediction approach in YOLO9000 (Redmon and Farhadi, 2017). The predicted coordinates are values relative to the location of each grid cell. In addition, each bounding box predicted by the network is defined as $\{p_x, p_y, p_w, p_h\}$, which is the actual output of the network during training. Given the center cell's coordinates (c_x, c_y) and width (w) and height (h) as the bounding box prior, the predictions correspond to:

$$b_x = \sigma(p_x) + c_x,$$
$$b_y = \sigma(p_y) + c_y,$$
$$b_w = p_w e^{p_w},$$
$$b_h = p_h e^{p_h},$$

where σ represents the sigmoid function that aims to limit the offset between 0 and 1. b_x and b_y denote the coordinates of the center point of the whole image. b_w and b_h represent the real width and height of the predicted boxes, respectively. b_x, b_y, b_w and b_h correspond to the predictions of the bounding box position. We also define the objectiveness of the prediction score $P_r(obj)$ (Redmon and Farhadi, 2018) in our method, which is predicted by logistic regression during training. For a prediction that has the largest overlap with a ground truth among all the predictions, the network trains the objectiveness

of the prediction score to be 1. Moreover, $P_r(obj)$ is also predicted to be 1 when one or more than one predicted bounding boxes overlap ground truth with $IoU > 0.5$. If neither of the above two conditions is met, $P_r(obj)$ learns to be 0, which means that the network ignores the predictions with $IoU < 0.5$ and not have the highest overlap with any ground truth. $IoU(b, obj)$ represents the relative overlap with the predicted bounding box and ground truth vehicle object. For a predicted bounding box b and a ground truth obj, the proportion of their overlap area is represented as $Area_c$, and the proportion of b and obj are represented as $Area_b$ and $Area_{obj}$, respectively. The IoU between bounding box b and ground truth obj can be calculated as:

$$IoU(b, obj) = \frac{Area_c}{Area_b + Area_{obj} - Area_c}.$$

The loss function for the entire vehicle detection network, i.e., L_{det}, can be divided into three components: (1) an objectiveness prediction loss L_{obj}, (2) a bounding box coordinate prediction loss L_{coord}, and (3) a category prediction loss L_{cls}. L_{det} is the weighted sum of these three losses. L_{det} is formulated as:

$$L_{det} = \lambda_{obj} L_{obj} + \lambda_{coord} L_{coord} + \lambda_{cls} L_{cls},$$

where we present binary cross-entropy loss function BCE on objectiveness prediction loss L_{obj}, category prediction loss L_{cls} and coordinate x, y prediction in loss L_{coord} calculation. A mean square loss function MSE is used on width w and height h prediction in loss L_{coord} calculation. The loss functions of BCE and MSE are calculated as follows:

$$BCE(\overline{x}_i, \overline{y}_i) = -w_i[\overline{y}_i \log \overline{x}_i + (1 - \overline{y}_i) \log(1 - \overline{x}_i)],$$

$$MSE(\overline{x}_i, \overline{y}_i) = (\overline{x}_i - \overline{y}_i)^2.$$

With the usage of BCE, the category prediction L_{cls} uses multiple label classifications instead of using softmax layers. Each logistic classifier is used oppositely for each class. The softmax loss is replaced by binary cross-entropy loss to achieve high performance in handling the label overlap relationships.

4. Experiments

4.1. Datasets

To evaluate the performance of our multi-scale adversarial network, we conduct vehicle detection from UAV imagery on the VisDrone dataset (Zhu et al., 2018) and UAVDT dataset (Du et al., 2018). DOTA (Xia et al., 2018), a satellite imagery dataset, is also involved in our experiments. The pretrained model is derived from the DOTA dataset with all categories trained on YOLOv3 (Redmon and Farhadi, 2018). The baseline experiments are conducted on the pretrained model fine-tuned on VisDrone and UAVDT dataset (detailed training setting of baselines are presented in Section 4.4). Our experiments focus on two categories: small vehicles and large vehicles, which are the only common categories annotated on DOTA (Xia et al., 2018), UAVDT dataset (Du et al., 2018) and VisDrone dataset (Zhu et al., 2018).

DOTA dataset: The DOTA dataset (Xia et al., 2018) consists of satellite imagery mainly collected from Google Earth. Some of the images are taken by satellite JL-1, and others are taken by satellite GF-2 derived from the China Center for Resources Satellite Data and Application. The ground truth is annotated with 15 common object categories and has a total of 188,282 instances. Given the large range in image sizes (from 800 × 800 to 4000 × 4000 pixels), we split large images and pad the small images to make all images have 1024 × 1024 pixels. In our adversarial network training, we choose DOTA images (5034 in total) that contain large vehicles and categories that correspond to small vehicle objects.

UAVDT dataset: The UAVDT dataset (Du et al., 2018) consists of UAV imagery with vehicles selected from 10-h long videos (80,000 representative frames). The images are fully annotated with bounding boxes with up to 14 kinds of attributes (e.g., weather condition, flying altitude, camera view, vehicle category, occlusion, etc.). The training set, val set, and testing set are comprised of 5000 images, 1658 images and 3316 images respectively, each with 1024 × 540 pixels. Since all images are selected from videos, the images from the same video are with similar backgrounds, camera views, and illumination (for images taken within a similar time period in a day).

VisDrone dataset: The VisDrone dataset (Zhu et al., 2018) consists of 288 video clips, formed by 261,908 frames and 10,209 static images. The video clips were captured by various drone-mounted cameras, covering a wide range of aspects with varying locations, environments, object types, and object density. The dataset was collected using various drone platforms, in different scenarios, and under various weather and lighting conditions. We selected images that contain large vehicles or small vehicles, and three subsets are divided according to the original division of in VisDrone benchmark dataset: train (6192 images), val (519 images), and test (1564 images). Since the images were collected via different platforms, the raw images are with two different sizes: 1360 × 765 pixels and 960 × 540 pixels.

4.2. Implementation details

All experiments are carried out on a workstation equipped with an Intel(R) Core(TM) i7-9800X CPU @ 3.80 GHz, two NVIDIA Geforce RTX 2080ti GPUs with 11G memory. The operating system is Ubuntu 16.04. We implement the whole program based on the publicly available Open MMLab Detection (Chen et al., 2019) framework built on the PyTorch platform. To obtain the pretrained model, we trained the split DOTA dataset from the YOLOv3 network in 30,000 steps with all 15 common object categories and an image size of 1024 × 1024 pixels. We resize the image to a size of 416 × 416 as input to the feature extractor network. Experiments involve different backbone networks that include ResNet-50 (He et al., 2016) and ResNeXt-101 (Xie et al., 2017).

The feature extractor network and discriminator in the multi-scale adversarial network are trained with the Stochastic Gradient Descent (SGD) optimizer and momentum of 0.9, weight decay of 0.0005, initial learning rate $lr_1 = 0.001$, $lr_2 = 0.001$ and $betas = (0.9, 0.999)$. The maximum iteration number is set 62,500 in all experiments. In order to improve the training efficiency, the learning rate lr_1 and lr_2 are attenuated according to the number of iterations in the training progress. When the iterations $itr = 50{,}000$ and $itr = 55{,}000$, the learning rate setting lr_1 and lr_2 reduced to 1/10 of the current value.

4.3. Evaluation metrics

In our experiments, we use the Mean Average Precision (mAP) to compare the performance among different models quantitatively. The mAP, calculated from the mean value of Average Precision (AP) of different categories, is widely used in the field of vehicle detection (Chen et al., 2019; Cai and Vasconcelos, 2018; Tian et al., 2019). Given m_{cls} as the number of categories, the values of AP and mAP are calculated as:

$$AP = \sum_{k=1}^{K} Pc(k) \Delta Rc(k)$$

$$mAP = \frac{1}{m_{cls}} \sum^{m_{cls}} AP$$

where Pc denotes the ratio of correctly predicted positive observations to the total predicted positive observations. Rc represents the ratio of correctly predicted positive observations to all observations in the actual class. AP is the average Pc value on the PR curve (taking Rc as the horizontal axis and Pc as the vertical axis) of detection results.

Furthermore, in order to evaluate the proposed method integrally, 6 metrics are used here: $AP_{[50,95]}$, AP_{50}, AP_{75}, AP_S, AP_M and AP_L. AP_{50}, AP_{75}, $AP_{[50,95]}$ denote mAP evaluated on $IoU = 0.5$, $IoU = 0.75$ and

Table 1
Comparisons of detection performance on UAVDT dataset.

Experiment	Backbone	$AP_{[50,95]}$	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	Params	Flops
Faster	ResNet-50	16.1	28.8	16.3	7.2	33.8	32.8	41.13	46.49
RetinaNet	ResNet-50	16.2	34.0	13.7	8.8	30.1	23.8	36.42	35.73
FCOS	ResNet-50	12.3	27.2	9.3	6.6	23.8	24.2	31.84	33.29
SABL	ResNet-50	16.9	29.1	17.5	7.7	34.8	34.0	41.91	106.09
Cascade	ResNet-50	16.8	30.2	16.7	7.6	34.7	35.1	68.93	74.29
SABL	ResNeXt-101	18.0	30.5	19.7	9.0	36.5	32.0	99.63	145.51
Cascade	ResNeXt-101	18.7	31.2	20.7	10.1	37.2	32.1	126.65	113.71
Baseline	Darknet-53	20.2	38.7	19.0	10.8	37.9	28.4	61.53	32.76
Baseline_Joint	Darknet-53	20.8	42.9	17.2	11.6	37.3	29.6	61.53	32.76
AdNet_SS	Darknet-53	21.3	42.6	18.3	11.5	37.7	30.7	77.16	32.76
AdNet_MS	Darknet-53	21.5	43.5	18.3	12.1	37.9	27.9	99.29	32.76

Table 2
Comparisons of detection performance on VisDrone dataset.

Experiment	Backbone	$AP_{[50,95]}$	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	Params	Flops
Faster	ResNet-50	22.0	39.1	22.4	2.2	32.2	55.6	41.13	46.49
RetinaNet	ResNet-50	22.7	44.3	20.7	4.5	31.3	53.5	36.42	35.73
FCOS	ResNet-50	16.6	32.8	15.1	2.8	21.9	49.6	31.84	33.29
SABL	ResNet-50	24.2	40.4	25.9	2.3	35.2	60.8	41.91	106.09
Cascade	ResNet-50	23.9	40.9	24.9	3.1	34.8	57.7	68.93	74.29
SABL	ResNeXt-101	25.0	41.2	26.9	2.3	36.3	61.7	99.63	145.51
Cascade	ResNeXt-101	26.0	43.1	28.0	2.9	37.6	62.4	126.65	113.71
Baseline	Darknet-53	29.7	55.7	29.1	11.7	39.7	57.1	61.53	32.76
Baseline_Joint	Darknet-53	31.6	58.4	31.0	12.9	42.5	59.7	61.53	32.76
AdNet_SS	Darknet-53	31.1	57.9	30.5	12.5	41.6	59.5	77.16	32.76
AdNet_MS	Darknet-53	32.3	60.4	31.1	13.0	42.7	59.6	99.29	32.76

$IoU = [0.50 : 0.05 : 0.95]$ respectively. AP, AP_S, AP_M and AP_L are mAP at different scales (Lin et al., 2014) with $IoU = [0.50 : 0.05 : 0.95]$. Specifically, AP_{50}, abbreviated as mean AP or mAP in different papers (Xia et al., 2018), is the most commonly used evaluation index among these 6 metrics.

4.4. Quantitative evaluation

We conduct a series of experiments on both the UAVDT and VisDrone dataset to showcase the advantages of the proposed architecture. To highlight the utility of the proposed method, the quantitative experiment results on both baselines and the proposed method are shown in Tables 1 and 2, where all training and testing processes use the same default runtime settings and image processing rules. The comparative experiments include not only widely used methods (*Faster* Ren et al., 2015, *RetinaNet* Lin et al., 2020, *Yolov3* Redmon and Farhadi, 2018), but also state-of-the-art approaches (*Cascade* Cai and Vasconcelos, 2018, *FCOS* Tian et al., 2019, *SABL* Wang et al., 2020). Since our proposed feature extractor network is based on *Yolov3* method, the experiment *Yolov3* is also considered as the *Baseline*. The experiment *AdNet_MS* lists results by our proposed multi-scale adversarial network.

Specifically, to ensure that the *Baseline* has the same access to the remote sensing data as *AdNet_MS*, we first train a pretrained model using remote sensing images (DOTA dataset in our experiment), and then derive detection model on the previous pretrained model. Furthermore, in order to show the experimental results under alternate training of the remote sensing images and UAV images, *Baseline_Joint* is set as another comparative experiment, which has the same feature extractor and training mechanism as *AdNet_MS*. The only difference between *Baseline_Joint* and *AdNet_MS* is that the *Baseline_Joint* discards the discriminator structure in Section 3.2, and no L_{adv} is calculated in loss function. Moreover, to highlight the advantage of the multi-scale design, we design a single-scale discriminator, named *AdNet_SS*. Compared to *AdNet_MS* with a multi-scale structure, *AdNet_SS* chooses the highest dimensional feature X_1 from the deep convolutional extractor, deprived of the second layer and third layer input.

We observe that the *AdNet_MS* network has the best performance in almost all precision metrics (up to 2.6% gains in $AP_{[50,95]}$ compared with *Baseline* and up to 10.3% gains in $AP_{[50,95]}$ compared with *Faster* based on ResNet-50) on both UAVDT Dataset and VisDrone Dataset, suggesting that our proposed multi-scale adversarial network improves the detection in UAV imagery with the integration of the domain adaptation and multi-scale design. It is worth noting that a few base networks show higher performance than the proposed method in AP_L. However, given the trade-off process of detecting large vehicles and small vehicles in UAV images via a certain model, the improvement of precision of small targets has a greater significance in overall detection evaluation AP.

To further analyze the complexity and computational efficiency of the proposed method, we present the Params (using M as a unit Chen et al., 2019) and Flops (using GFLOPs as a unit Chen et al., 2019) in last two columns in Tables 1 and 2. We observe an increased number of parameters in *AdNet_MS* compared with other models, while the counts of Flops in *AdNet_MS* remain to be the lowest (unchanged compared with *Baseline*). These results are presumably due to the attachment of the multi-scale discriminator in the proposed algorithm, which leads to more parameters in the model but results in no additional calculation during the detection phase. Although the proposed method has more parameters, considering the low computation in the detection phase, the improvement of the general accuracy, and the improvement of detection, we believe the advantages of attaching multi-scale discriminator outweigh the disadvantages.

4.5. Qualitative evaluation

Selected experimental results generated by our proposed multi-scale adversarial network are shown in Figs. 6 and 7, where the detected objects are labeled manually with green rectangles with predicted categories. The first, second, and third rows of images in Figs. 6 and 7 are examples of vehicle detection results respectively from the UAVDT dataset and the VisDrone dataset. The presented examples are selected to cover day, night, and fog conditions and contain vehicles objects from various viewpoints and altitudes. We observe that most vehicles in these images are accurately detected, demonstrating the

Fig. 6. The detection results in UAVDT dataset by the proposed multi-scale adversarial network. The detected vehicles are labeled with green rectangles and marked with categories and confidence scores. The last row shows selected images with wrongly detected or undetected vehicle objects. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

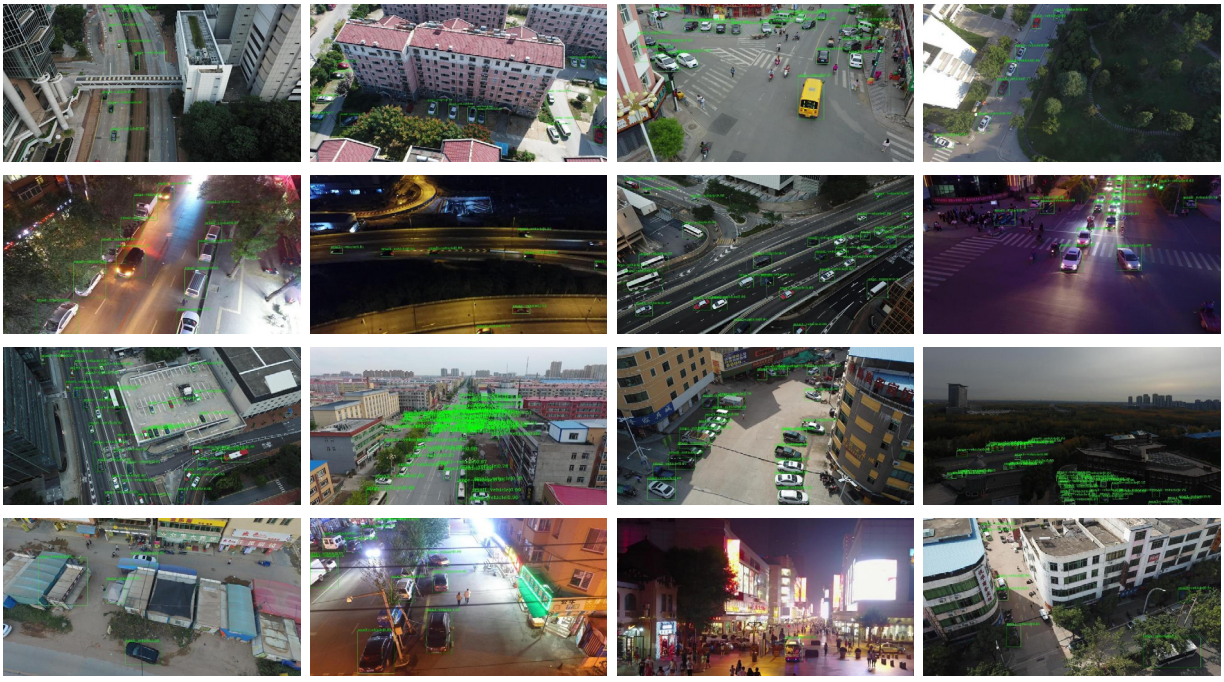

Fig. 7. The detection results in the VisDrone dataset by the proposed multi-scale adversarial network. The detected vehicles are labeled with green rectangles and marked with categories and confidence scores. The last row shows selected images with wrongly detected or undetected vehicle objects. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

great capability of our algorithm in detecting vehicles objects from UAV imagery.

The last row of images in Figs. 6 and 7 are images with wrongly detected and undetected objects. We observe that most wrong detections appear in images taken from a high altitude and images that contain vehicles greatly differing in size, indicating that the proposed detectors can be further improved. We further observe that many occluded vehicles by trees, buildings, and poles are missed (see the last row in Fig. 7), suggesting that detecting occluded objects remains to be a challenging task. We also notice the existence of some wrongly annotated ground truths, like the large green rectangular in the first image in the last row in Fig. 6. We acknowledge that wrongly labeled ground truths affect model training and experiment evaluation. However, it is difficult to correct all the labels in these training datasets.

Fig. 8 presents the feature visualization from randomly selected images in the testing set on the VisDrone dataset. As described in

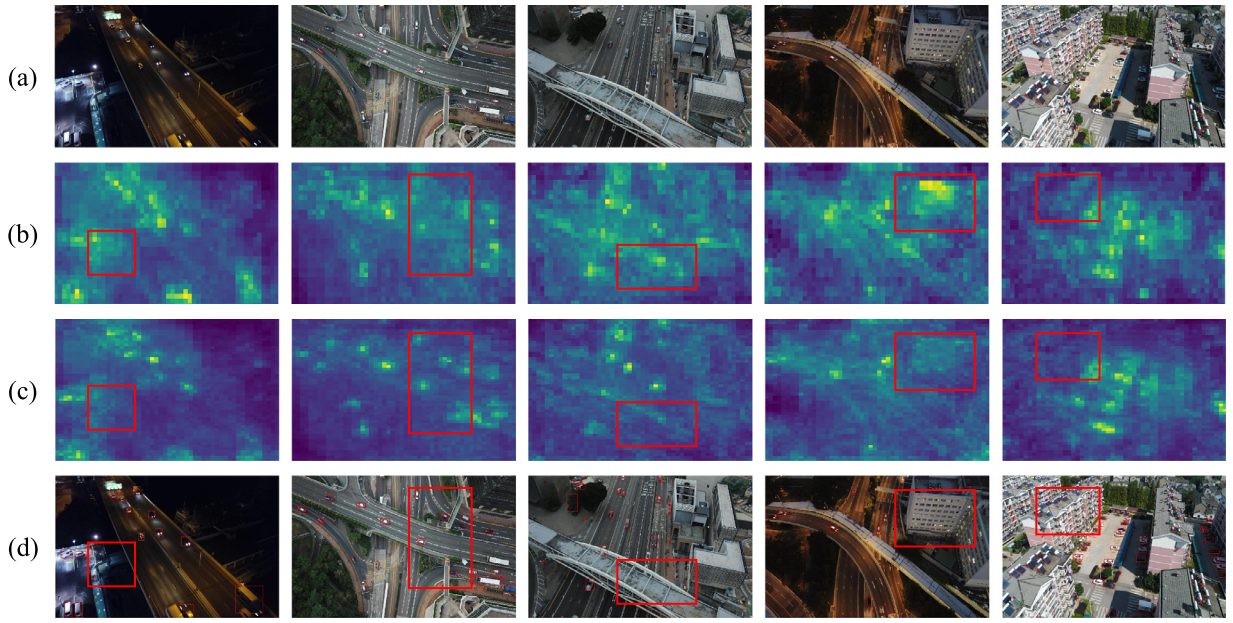

Fig. 8. Feature visualization from the *Baseline* and the proposed $AdNet\_MS$. (a) randomly selected images in the VisDrone dataset; (b) visualization results from the *Baseline* model; (c) visualization results from the $AdNet\_MS$ model; (d) ground truth labeled images. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Section 3.2, our proposed method aims to learn a function G_e to obtain collaborative features $G_e(x_u)$ beyond perspectives and altitudes in UAV image x_u. Thus, features $G_e(x_u)$ and features from the *Baseline* model explicitly reflect the functionality of the designed multi-scale adversarial network. The rows (a) and (d) in Fig. 8 respectively show the original images and images labeled with ground truth by red rectangles. Row (b) and (c) respectively present the feature visualization from the *Baseline* model and the $AdNet\_MS$ model. We use interpolation and element-sum operation to combine multi-scale features. All features are presented in the HSV color space using channel maximum squeeze. From Fig. 8, we observe that the highly responsive regions in the extracted feature map from the proposed $AdNet\_MS$ contain clearer structures of vehicles than the *Baseline* model that often fails to discriminate vehicles from the background (see the highlighted red rectangle).

Furthermore, for the purpose of qualitative comparison between our proposed networks and baseline, we present randomly selected examples of detection results on *Baseline* and $AdNet\_MS$ in Fig. 9. Note that (a) and (b) show examples from the UAVDT dataset, while (c) and (d) show examples from the VisDrone dataset. Additionally, selected areas are zoomed in with objects in multiple scales (examples in the first line in Fig. 9), occlusion (second line in Fig. 9), and poor photograph conditions (third line in Fig. 9) on each detection result, where the improvement of detection results based on our proposed network is visible. Compared with detection results in *Baseline*, multi-scale discriminator benefits the whole $AdNet\_MS$ in aligning features under different perspectives and altitudes, and the results demonstrate that our proposed $AdNet\_MS$ improves generalization of vehicle detection in UAV imagery and is capable of detecting vehicles with multiple scales, with occlusion, and under poor photograph conditions.

4.6. Ablation analysis

To examine the functionality of the discriminator, we set up a series of ablation experiments governed by λ_{adv}, a parameter to balance the detection loss L_{det} and adversarial loss L_{adv}. Tables 3 and 4 present the results with different λ_{adv}. First, we observe two obvious AP improvements in the middle of training, corresponding to the reduction in learning rates. The results demonstrate that our learning rate setting is effective, suggesting that an attenuated learning rate facilitates training efficiency and post-stability. We further observe that, as the λ_{adv} increases from 1 to 2 in Table 3 and from 2 to 4 in Table 4, the performance of the adversarial network increases, which verifies the significance of multi-scale discriminator, and when λ_{adv} increases from 4 to larger values, however, the performance decreases. In the designed adversarial architecture, λ_{adv} aims to grant the extractor the ability to confuse the discriminator, facilitating the extractor to acquire deep common features from both the remote sensing dataset and UAV dataset. As λ_{adv} increases, the discriminator has an increasing influence on the feature extractor. With λ_{adv} increasing to a certain point, however, it makes the deep extractor consider more on discriminator and less on extracted features for detection, which worsens the performance as shown in the above experiments. In our comparative experiments, we obtain the results from our proposed multi-scale adversarial network with $\lambda_{adv} = 2$ and $\lambda_{adv} = 4$ on UAVDT dataset and VisDrone dataset. The results above prove the crucial role of the discriminator in the performance of our network, as it gives positive feedback during the training of the extractor.

Table 3
Comparisons of detection performance with different λ_{adv} on UAVDT dataset.

Experiment	$AP_{[50,95]}$	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Baseline_Joint	20.8	42.9	17.2	11.6	37.3	29.6
$AdNet\_MS_1$	21.4	43.1	18.3	12.3	36.7	30.0
$AdNet\_MS_2$	21.5	43.5	18.3	12.1	37.9	27.9
$AdNet\_MS_4$	21.1	41.6	19.0	11.8	37.8	28.0
$AdNet\_MS_6$	20.7	41.5	18.2	11.7	36.1	32.0

Table 4
Comparisons of detection performance with different λ_{adv} on VisDrone dataset.

Experiment	$AP_{[50,95]}$	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Baseline_Joint	31.6	58.4	31.0	12.9	42.5	59.7
$AdNet\_MS_2$	32.0	59.4	31.5	12.9	42.5	59.5
$AdNet\_MS_4$	32.3	60.4	31.1	13.0	42.7	59.6
$AdNet\_MS_6$	32.3	60.0	31.5	12.9	42.9	59.6
$AdNet\_MS_{10}$	31.5	58.9	31.0	12.9	42.2	60.3

Fig. 9. Qualitative comparisons of detection results based on the proposed multi-scale adversarial network and baseline. From left to right: (a) detection result of *Baseline* from UAVDT dataset; (b) detection result of *AdNet_MS* from UAVDT dataset; (c) detection result of *Baseline* from VisDrone dataset; (d) detection result of *AdNet_MS* from VisDrone dataset. Compared with results from *Baseline*, the proposed method detects more vehicles in multiple scales, occlusion and poor photography conditions.

Table 5
Quantitative results of different situation in UAVDT dataset.

Experiment	Day	Night	Fog	Low	Medium	High	Front	Side	Bird
Baseline	19.1	33.4	12.4	33.0	23.1	10.1	21.4	24.1	15.5
Baseline_Joint	20.7	32.6	11.3	35.9	22.8	10.9	22.8	25.8	14.5
AdNet_MS	20.9	34.5	11.6	37.6	23.3	11.3	23.7	26.1	15.1

4.7. Quantitative analysis on different attributes

In previous sections, we have shown that the proposed multi-scale adversarial network achieves great performance in both UAVDT dataset and VisDrone dataset. To further explore the performance variation with different image shooting conditions, we conduct experiments using the annotation of attributes in the UAVDT dataset, which include *Weather Condition*, *Flying Altitude*, *Camera View*. **Weather Condition** (including *Day*, *Night* and *Fog*) determines the illuminating condition when images are being captured, which affects the appearance representation of objects. **Flying Altitude** (including *Low*, *Medium* and *High* altitude) represents the flying height of UAVs, leading to scale variation of objects. **Camera View** consists of 3 object views, i.e., *Front*, *Side* and *Bird* view.

The attributes of UAV images suggests the great heterogeneity of images in the UAVDT dataset, which leads to challenges for vehicle detection but benefits the model generalization capability. Table 5 presents the $AP_{[50,95]}$ of vehicle detection results from *Baseline*, *Baseline_Joint*, *AdNet_SS*, *AdNet_MS* with different imaging situations. Compared with the results from the *Baseline* and *Baseline_Joint*, the proposed adversarial network achieves improved performance in all situations except *Fog* and *Bird* view. The results suggest that our network achieves more robustness to variation in appearance in UAV imagery. Despite the good performance of the baseline in images of *Fog* and *Bird*, its performance on images of other attributes is very limited (presumably resulting from overfitting). As our network introduces satellite imagery into the UAV imagery by facilitating their feature alignment in the feature space, it is able to improve detection performance on various sensor altitudes and background conditions.

From the model performance under different *Flying Altitude*, we notice that the proposed *AdNet_MS* network achieves the highest $AP_{[50,95]}$ in *Low*, *Medium* and *High*, compared with other models. Specifically, all models perform better in *Day* and *Night* situations, compared with situations of *Fog* situations. This can be explained by the fact that fog scenes generally present scarce texture information and little sharp details. In addition, we find that models present different performances given images taken at different flying altitudes. All models show better performance on images taken at lower flying altitudes, as they contain more detailed information that benefits the object detector. Camera view also plays an important role in the detector performance, with side view images showing the highest detection accuracy, presumably due to their diverse details.

5. Conclusions

The large differences in vehicle appearance and size and the complex and diverse scenes from UAV images have caused great difficulty in the vehicle detection tasks. Such variation presents a significant challenge in particular to supervised learning-based approaches that rely on training examples that are representative of all possible scenarios. In this study, we seek to address the large variation in UAV imagery from a novel perspective. We argue that, while there might be a large variation in appearance in UAV imagery, there is much less in traditional remote sensing imagery, such as the ones taken from satellite or aircraft. Thus, satellite imagery can serve as a common anchor or reference for aligning the UAV features. The rich training remote sensing images further contribute to the robustness of such anchor training.

Following this rationale, we propose a novel multi-scale adversarial network for improved vehicle detection in UAV imagery. Our proposed framework consists of three sub-networks: a deep convolutional feature extractor, a multi-scale discriminator, and a vehicle detection network. The feature extractor is a two-path structure with one path for the UAV imagery and another for the satellite imagery. The shared weights in feature extractor allow us to exploit the large collections of labeled remote sensing imagery for improved vehicle detection in UAV imagery. The multi-scale discriminator performs feature alignment by forcing the extractor to learn features that are indistinguishable as to whether they are from UAV or remote sensing imagery, in an adversarial manner.

We demonstrate the effectiveness of our approach on the UAVDT and VisDrone datasets. Experimental results suggest that our proposed algorithm improves the vehicle detection accuracy in both datasets, and the proposed model achieves great performance in images taken from different perspectives, at different altitudes, and under different imaging situations.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grants 42090012, 61671332, 41771452, 51708426, 41890820 and 41771454, the Natural Science Foundation of Hubei Province in China under Grant 2018CFA007, the Independent Research Projects of Wuhan University under Grant 2042018kf0250.

References

Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., 2014. Domain-adversarial neural networks. arXiv preprint arXiv:1412.4446. 1050.

Al-Moslmi, T., Omar, N., Abdullah, S., Albared, M., 2017. Approaches to cross-domain sentiment analysis: A systematic literature review. IEEE Access 5, 16173–16192.

Aslani, S., Mahdavi-Nasab, H., 2013. Optical flow based moving object detection and tracking for traffic surveillance. Int. J. Electr. Comput. Energ. Electron. Commun. Eng. 7 (9), 1252–1256.

Audebert, N., Le Saux, B., Lefèvre, S., 2018. Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks. ISPRS J. Photogramm. Remote Sens. 140, 20–32.

B, C.Z.A., C, P.M.A., D, C.G., E, Z.W., F, M.D., D, F.G., 2020. Identifying and mapping individual plants in a highly diverse high-elevation ecosystem using UAV imagery and deep learning. ISPRS J. Photogramm. Remote Sens. 169, 280–291.

Bak, S., Corvee, E., Bremond, F., Thonnat, M., 2010. Person re-identification using haar-based and DCD-based signature. In: IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). pp. 1–8.

Benjamin, K., Diego, M., Devis, T., 2018. Detecting mammals in UAV images: Best practices to address a substantially imbalanced dataset with deep learning. Remote Sens. Environ. 216, 139–153.

Bousmalis, K., Trigeorgis, G., Silberman, N., Krishnan, D., Erhan, D., 2016. Domain separation networks. In: Advances in Neural Information Processing Systems. pp. 343–351.

Cai, Z., Vasconcelos, N., 2018. Cascade r-cnn: Delving into high quality object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6154–6162.

Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., Zhang, Z., Cheng, D., Zhu, C., Cheng, T., Zhao, Q., Li, B., Lu, X., Zhu, R., Wu, Y., Dai, J., Wang, J., Shi, J., Ouyang, W., Loy, C.C., Lin, D., 2019. MMDetection: Open MMLab detection toolbox and benchmark. arXiv preprint arXiv:1906.07155.

Chen, X., Xiang, S., Liu, C.-L., Pan, C.-H., 2014. Vehicle detection in satellite images by hybrid deep convolutional neural networks. IEEE Geosci. Remote Sens. Lett. 11 (10), 1797–1801.

Chen, Y.-C., Zheng, W.-S., Lai, J.-H., Yuen, P.C., 2016. An asymmetric distance model for cross-view feature mapping in person reidentification. IEEE Trans. Circuits Syst. Video Technol. 27 (8), 1661–1675.

Colomina, I., Molina, P., 2014. Unmanned aerial systems for photogrammetry and remote sensing: A review. ISPRS J. Photogramm. Remote Sens..

Dai, J., Li, Y., He, K., Sun, J., 2016. R-fcn: Object detection via region-based fully convolutional networks. In: Advances in Neural Information Processing Systems. pp. 379–387.

Du, D., Qi, Y., Yu, H., Yang, Y., Duan, K., Li, G., Zhang, W., Huang, Q., Tian, Q., 2018. The unmanned aerial vehicle benchmark: Object detection and tracking. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 370–386.

Gevaert, C.M., Suomalainen, J., Tang, Kooistra, L., 2015. Generation of spectral–temporal response surfaces by combining multispectral satellite and hyperspectral UAV imagery for precision agriculture applications. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 8 (6), 3140–3146.

Girshick, R., 2015. Fast R-CNN. Comput. Sci..

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2015. Region-based convolutional networks for accurate object detection and segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 38 (1), 142–158.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. In: Advances in Neural Information Processing Systems. pp. 2672–2680.

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 2961–2969.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778.

Huang, R., Zhang, S., Li, T., He, R., 2017. Beyond face rotation: Global and local perception gan for photorealistic and identity preserving frontal view synthesis. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 2439–2448.

Hung, W.-C., Tsai, Y.-H., Liou, Y.-T., Lin, Y.-Y., Yang, M.-H., 2018. Adversarial learning for semi-supervised semantic segmentation. In: Proceedings of the British Machine Vision Conference (BMVC). p. 65.

Jabar, F., Farokhi, S., Sheikh, U., 2015. Object tracking using SIFT and KLT tracker for UAV-based applications. In: 2015 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS). IEEE, pp. 65–68.

Kalantar, B., Mansor, S.B., Halin, A.A., Shafri, H.Z.M., Zand, M., 2017. Multiple moving object detection from UAV videos using trajectories of matched regional adjacency graphs. IEEE Trans. Geosci. Remote Sens. 1–16.

Kamate, S., Yilmazer, N., 2015. Application of object detection and tracking techniques for unmanned aerial vehicles. Procedia Comput. Sci. 61, 436–441.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems. pp. 1097–1105.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al., 2017. Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4681–4690.

Li, M., Li, D., Fan, D., 2012. A study on automatic UAV image mosaic method for paroxysmal disaster. Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. 39, 123–128.

Li, W., Li, F., Luo, Y., Wang, P., 2020. Deep domain adaptive object detection: a survey. In: IEEE Symposium Series on Computational Intelligence. pp. 1808–1813.

Li, Q., Mou, L., Xu, Q., Zhang, Y., Zhu, X.X., 2019. R3-Net: A deep network for multioriented vehicle detection in aerial images and videos. IEEE Trans. Geosci. Remote Sens. 57 (7), 5028–5042.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2020. Focal loss for dense object detection. IEEE Trans. Pattern Anal. Mach. Intell. 42 (2), 318–327.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L., 2014. Microsoft coco: Common objects in context. In: Proceedings of the European Conference on Computer Vision (ECCV). Springer, pp. 740–755.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C., 2016. SSD: Single shot multibox detector. In: Proceedings of the European Conference on Computer Vision (ECCV). Springer, pp. 21–37.

Liu, J., Shah, M., Kuipers, B., Savarese, S., 2011. Cross-view action recognition via view knowledge transfer. In: CVPR 2011. IEEE, pp. 3209–3216.

Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y., 2018. Spectral normalization for generative adversarial networks. In: International Conference on Learning Representations (ICLR).

Moranduzzo, T., Melgani, F., 2014. Detecting cars in UAV images with a catalog-based approach. IEEE Trans. Geosci. Remote Sens. 52 (10), 6356–6367.

Najibi, M., Rastegari, M., Davis, L.S., 2016. G-cnn: an iterative grid based object detector. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2369–2377.

Pang, J., Chen, K., Shi, J., Feng, H., Ouyang, W., Lin, D., 2019. Libra R-CNN: Towards balanced learning for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Peng, X., Yu, X., Sohn, K., Metaxas, D.N., Chandraker, M., 2017. Reconstruction-based disentanglement for pose-invariant face recognition. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 1623–1632.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 779–788.

Redmon, J., Farhadi, A., 2017. YOLO9000: better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7263–7271.

Redmon, J., Farhadi, A., 2018. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems. pp. 91–99.

Rodríguez-Canosa, G.R., Thomas, S., Del Cerro, J., Barrientos, A., MacDonald, B., 2012. A real-time method to detect and track moving objects (DATMO) from unmanned aerial vehicles (UAVs) using a single camera. Remote Sens. 4 (4), 1090–1111.

Shen, Z., Liu, Z., Li, J., Jiang, Y.-G., Chen, Y., Xue, X., 2017. Dsod: Learning deeply supervised object detectors from scratch. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 1919–1927.

Shermin, T., Lu, G., Teng, S.W., Murshed, M., Sohel, F., 2020. Adversarial network with multiple classifiers for open set domain adaptation. IEEE Trans. Multimed..

Taigman, Y., Polyak, A., Wolf, L., 2017. Unsupervised cross-domain image generation. In: International Conference on Learning Representations (ICLR).

Tan, M., Pang, R., V.Le, Q., 2020. Efficientdet: Scalable and efficient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Tian, Z., Shen, C., Chen, H., He, T., 2019. FCOS: Fully convolutional one-stage object detection. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 9626–9635.

Wang, J., Zhang, W., Cao, Y., Chen, K., Pang, J., Gong, T., Shi, J., Loy, C.C., Lin, D., 2020. Side-aware boundary localization for more precise object detection. In: Proceedings of the European Conference on Computer Vision (ECCV).

Xia, G.-S., Bai, X., Ding, J., Zhu, Z., Belongie, S., Luo, J., Datcu, M., Pelillo, M., Zhang, L., 2018. DOTA: A large-scale dataset for object detection in aerial images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3974–3983.

Xie, S., Girshick, R.B., Dollár, P., Tu, Z., He, K., 2017. Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5987–5995.

Xu, B., Wang, N., Chen, T., Li, M., 2015. Empirical evaluation of rectified activations in convolutional network. arXiv preprint arXiv:1505.00853.

Yang, C., Fan, B., Ji, L., Luo, J., Yu, H., 2014. Speeded up low-rank online metric learning for object tracking. IEEE Trans. Circuits Syst. Video Technol. 25 (6), 1.

Yoo, D., Park, S., Lee, J.-Y., Paek, A.S., So Kweon, I., 2015. Attentionnet: Aggregating weak directions for accurate object detection. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 2659–2667.

Zhang, C., Wu, L., Wang, Y., 2019. Crossing generative adversarial networks for cross-view person re-identification. Neurocomputing 340, 259–269.

Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., Metaxas, D.N., 2017a. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 5907–5915.

Zhang, Y., Yuan, X., Fang, Y., Chen, S., 2017b. UAV low altitude photogrammetry for power line inspection. ISPRS Int. J. Geo Inf. 6 (1), 14.

Zheng, W., Hu, R., Chao, L., Yi, Y., Leng, Q., 2015. Zero-shot person re-identification via cross-view consistency. IEEE Trans. Multimed. 18 (2), 1.

Zhou, G., Ambrosia, V., Gasiewski, A.J., Bland, G., 2009. Foreword to the special issue on unmanned airborne vehicle (UAV) sensing systems for earth observations. IEEE Trans. Geosci. Remote Sens. 47 (3), 687–689.

Zhou, H., Wei, L., Lim, C.P., creighton, d., Nahavandi, S., 2018. Robust vehicle detection in aerial images using bag-of-words and orientation aware scanning. IEEE Trans. Geosci. Remote Sens. 56 (12), 7074–7085.

Zhu, J.-Y., Park, T., Isola, P., Efros, A.A., 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 2223–2232.

Zhu, P., Wen, L., Du, D., Bian, X., Ling, H., Hu, Q., Nie, Q., Cheng, H., Liu, C., Liu, X., et al., 2018. Visdrone-det2018: The vision meets drone object detection in image challenge results. In: Proceedings of the European Conference on Computer Vision (ECCV), Workshops, Vol. 11133. pp. 437–468.