

Airline Baggage Appearance Transportability Detection Based on A Novel Dataset and Sequential Hierarchical Sampling CNN Model

QINGJI GAO AND PEIWEN LIANG 

Robotics Institute, Civil Aviation University of China, Tianjin 300300, China

Corresponding author: Peiwen Liang (liang_pw@163.com)

ABSTRACT Self-service bag drop efficiently assists passengers to check-in their baggage in the airport. Nevertheless, the baggage appearance transportability cannot be accurately detected by existing self-service bag drop equipment. We plan to adopt a convolutional neural network with video input to detect the appearance transportability of baggage. However, public baggage picture datasets are captured in the daily background, thus existing approaches trained on these datasets achieve imprecise performance for airport self-service bag drop. We introduce a new dataset for airport self-service bag drop named ASS-BD and a novel sequential hierarchical sampling multi-object tracker. Most of the video clips that comply with the consignment regulations were recorded in the airport scene. Video clips that do not comply with the consignment regulations were recorded in the laboratory simulation scene. A sequential hierarchical sampling multi-object tracking baseline is adopted to solve some problematic frames due to part occlusion, rare pose, and motion blur. We conduct experiments to demonstrate that our dataset is suitable for the airport self-service bag drop scenario. Our approach is capable of the inspection task of air baggage appearance transportability in real-time.

INDEX TERMS Airport self-service bag drop dataset, airline baggage appearance transportability inspection, anchor-free object detection, multi-object tracking.

I. INTRODUCTION

A. BACKGROUND

Airport Self-service bag drop can reduce the check-in time of passengers, improve the passenger experience, and maximize the throughput of terminal passengers, which is an essential means to simplify the check-in process. Detecting the transportability of baggage is the key to self-service bag drop for determining whether the passenger's bag meets the check-in conditions. With the rapid development of artificial intelligence and human-computer interaction, we have developed self-service bag drop equipment which is used to check whether the air baggage meets the check-in conditions and applied them to Beijing Daxing International Airport, Guangzhou Baiyun International Airport, and Tianjin Binhai International Airport.

Self-service bag drop could detect whether the passenger's baggage meets the check-in regulations from the International


The associate editor coordinating the review of this manuscript and approving it for publication was Danilo Pelusi .

TABLE 1. Elements of the appearance transportability.

A	B	C	D	E
Tag	Pallet	One	Backpack	Suitcase

Air Transport Association, so as to be accepted to check-in. The regulations of baggage transportability are about the weight, size, and "appearance transportability". The "appearance transportability" checks the type and number of baggage, whether the baggage is tied with a tag, and whether the backpack is equipped with a pallet. Table 1 shows the elements that need to be judged for the "appearance transportability" of baggage. The symbol "&" means that it satisfies two elements at the same time. Therefore, the baggage appearance that meets the transportability regulations is A&B&C&D, A&B&C&E, A&C&E. In the following, "appearance transportability" will be used to represent meeting one of the above regulations.

A variety of self-service bag drop equipment has been used in airports, and many key technologies have been

explored along with the development of equipment. In 2012, Bagdrop's [1] latest product was widely promoted and applied at Amsterdam Airport. The system is equipped with a barcode scanner, weighing sensor, and camera, which can support baggage weighing, tag verification, capture the appearance of baggage, and use 3D imaging technology to obtain a three-dimensional model of the baggage. In 2015, the self-service bag drop developed by ICM Airport Technics [2] was equipped with multiple three-dimensional scanners to scan baggage simultaneously, which can analyze the size, placement, and appearance of the baggage. In 2018, the CHECKITXPRESS [3] self-service bag drop system, which was jointly developed by the Innovative Travel Solutions (ITS) of Vancouver International Airport and a New Zealand airport baggage handling company, Glidepath used innovative camera technology to improve the speed and efficiency of tag information extraction. In 2019, a self-service bag drop system jointly developed by the Civil Aviation University of China and Tianjin Hangda Aviation Equipment company was installed at Beijing Daxing International Airport [4]. As shown in Fig. 1, the system uses comprehensive 3D imaging technology and extensive smart barcode recognition technology to detect the baggage appearance by extracting three-dimensional point cloud features, including detection of pallets, multiple baggage, and baggage with irregular surfaces. The purpose of these devices is to improve the accuracy of the bag appearance transportability detection.

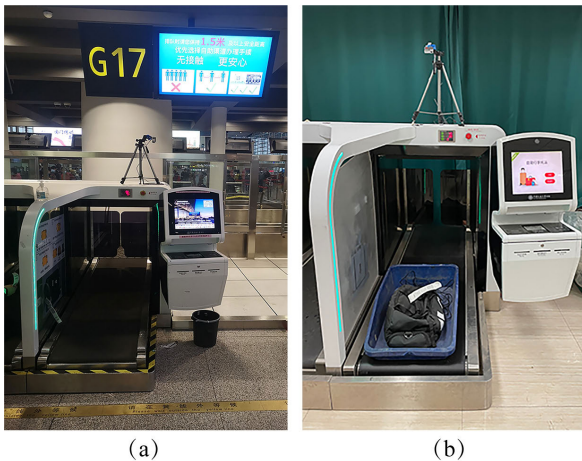


FIGURE 1. Self-service bag drop equipment. Picture (a) was captured in the applied airport scene, picture (b) was captured in the laboratory simulation scene. The camera holder is 47 cm high, and the camera angle is about 45 degrees horizontally downward.

Although the existing self-service bag drop technology [5] in airports has made some breakthroughs, they cannot detect strictly the appearance transportability regulations (A&B&C&D, A&B&C&E, A&C&E) listed before Table 1. As shown in Fig. 2, in the practical use of the equipment, the detection method based on static point cloud can detect the “easy” placement of multiple pieces of baggage, but it can do nothing for some “hard” samples. Moreover, this method of extracting geometric features based on the top view

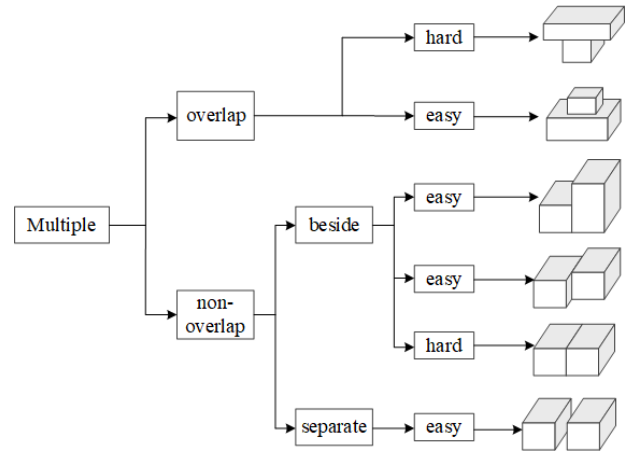


FIGURE 2. The ways of place multiple pieces of baggage, “hard” denotes this way is challenging to detect the number of baggage, “easy” denotes this way is easy to detect the number of baggage.

point cloud map of the baggage and then classifying it often misidentifies the type of baggage, so that passengers must put baggage into a pallet to check it, which seriously affects the passenger experience and increases unnecessary time. When passengers use the self-service bag drop, there will be complicated human-computer interaction behaviors and a wide variety of baggage, so this detection method based on a static three-dimensional point cloud cannot meet rigorous and efficient requirements. To sum up, the existing detection methods based on static point cloud cannot accurately detect the category and number of baggage, which cause loss of baggage. However, the detection algorithm based on deep learning has fast detection speed, low false detection rate, and strong versatility, we monitor the entire process of passengers’ checked baggage and adopt a video-based real-time multi-category multi-target tracking deep learning algorithm to detect the appearance transportability. In the applicable airport scenario, the detection task of appearance transportability faces several challenges: the complex background of the detection area in the video, the different forms of passenger check-in operations, different baggage, different pallets, different positions of tied tags, the interference from the baggage of queuing passengers, and the lack of proper actual self-service bag drop video dataset.

B. CONTRIBUTION

Firstly, a new video dataset ASS-BD is provided for detecting the appearance transportability, which consists of suitcases, backpacks, pallets, and tags. As far as we know, this is the first multi-target tracking video dataset tailored to the context of the airport self-service bag drop tasks. As shown in Fig. 1(a), the passengers’ checking baggage videos were recorded in the applicable airport scenario. As shown in Fig. 1(b), we recorded some videos of different volunteers that did not meet the appearance transportability in the laboratory simulation scene. Secondly, we propose a real-time model benchmark called sequential hierarchical sampling multi-object tracker based on the Centernet [6]

detection framework. The detection performance of this model on the ASS-BD dataset is better than the original Centernet. In addition, our method has better tracking performance than YOLOv3-deepsort [7].

C. PAPER ORGANIZATION

This paper is organized as follows: In section II, we provide recent related works of detecting appearance transportability approaches, object detection, and multi-object tracking. Section III introduces Airport Self-Service Bag Drop ASS-BD dataset. In section IV, we describe the architecture of the appearance transportability detection network. In section V, we introduce experiments and report the analyzed results. And in section VI, we summarize the findings and contributions of this study.

II. RELATED WORK

A. CONVENTIONAL APPEARANCE TRANSPORTABILITY DETECTION APPROACHES

Baggage is usually checked manually for compliance with the appearance transportability, some researches [5] on 3D detection attempt to use the point cloud generated by depth cameras to solve this problem. The proposed counting method uses image segmentation based on the height map which is projected by scanned baggage 3D point cloud, but the whole 3D point cloud information of placing the baggage on the conveyor belt will not be obtained, so the “hard” overlap placement method in Fig. 2 cannot correctly detect the amount of baggage. As shown in Fig. 2, the “hard” beside placement is not detected correctly by this approach that is overwhelmingly dependent on the height map projected by the 3D point cloud. These methods could accurately detect the dimensions of baggage and make human-computer interaction more natural. However, they are all vulnerable to the completeness of static 3D point cloud information scanned by depth cameras. Hence, this method is not proper to detect the appearance transportability of baggage.

B. OBJECT DETECTION AND MULTI-OBJECT TRACKING METHODS BASED ON DEEP LEARNING

Deep learning improves the ability of models to express the feature; many object detection and multi-object tracking technologies based on deep learning have been explored to complete detection tasks in different application scenarios. A practical autonomous driving system urges the need to reliably and accurately detect vehicles and persons, researchers who participated in the 2D detection track of the Waymo Open Dataset Challenges integrate both popular two-stage detector and one-stage detector with anchor free fashion to yield a robust detection on the Waymo Open Dataset v1.2 [8]. For smart city applications of tracking multiple targets across multiple cameras, MTMCT [9] framework that imposes a trajectory-based camera link model for vehicle re-identification(re-ID) could effectively and reliably track a

wide range of targets. This section briefly reviews existing object detection and multi-object tracking works.

There are two categories of object detection algorithms based on deep learning: the anchor-based detector and the anchor-free detector. Faster R-CNN [10] and YOLOv3 [11] are the classic methods of the anchor-based detector. The two-stage method Faster R-CNN firstly generate the candidate area of the target in the image and then classify the target bounding box, whose detection accuracy higher but slower. The one-stage detector YOLOv3 treats the detection problem as a regression problem, it predicts bounding boxes and class probabilities directly from full images in one evaluation. This type of method is fast but has insufficient detection accuracy of small targets. Cornernet [12] and Centernet [6] are the classic methods of the anchor-free detector, Cornernet first predicts the possible paired keypoints of the objects, and then finds matching corner points belonging to the same object by predicting an embedding vector, finally localize corners of bounding boxes through corner-pooling. Centernet detects an object bounding box as the center point and then uses this predicted center to find coordinates and offsets of the bounding box. Since the anchor-based algorithm needs to set many hyper parameters for anchors, and most anchors are negative samples during the training process, which causes a waste of computing resources. The center-free detection method Centernet is more suitable for the research background of self-service bag drop in this paper, it could deal with some part occlusion and large-angle posture transformations of objects, and realize real-time detection with better accuracy.

There will be some baggage obstructing each other in the video dataset, tags being obscured by baggage, and motion blur situation. In this way, it is very likely that missed and false detections occur in the key areas of detecting the appearance transportability of baggage, leading to the ultimate result is not accurate. The features of target motion trajectory could be automatically analyzed and extracted by multi-object tracking, which makes up the lack of object detection. The detection-based multi-object tracking is widely used in intelligent video surveillance scenarios. FairMOT [13] based on the Centernet combines the two tracking key components, object detection and identity re-identification module (re-ID), into the same network to accurately extract re-ID features for achieving excellent multi-object tracking results. Therefore, this paper improves some network structures on the basis of FairMOT so that it can meet the inspection task of appearance transportability in real scenarios.

C. OBJECT DETECTION DATASET

Many datasets were established for object detection, such as ImageNet [14], MSCOCO [15], and Google Open Image v4 [16], whose images are not video sequences with self-service bag drop scenarios. ImageNet dataset contains 14197122 images with 21841 categories, and MSCOCO dataset consists of 328000 images with 91 categories, Google Open Image v4 contains 1.74M bounding boxes of

600 categories. Meanwhile, there are multi-object tracking datasets like MOT16 [17], UA-DETRARC [18], and Vis-Drone [19], whose images are pedestrians, vehicles, and bicycles all captured in life scenes. We will select backpacks and suitcases images from the Google Open Image v4 dataset and name them "supplemental baggage" to increase the diversity of the video dataset ASS-BD. All datasets mentioned above are not proper for appearance transportability detection due to they are just individual pictures that belong to target categories in life scenes or video datasets without target categories. They cannot meet the requirements of the self-service bag drop task, which not only to detect targets but also to track targets over a long period of time. Therefore, we need to record the self-service bag drop video dataset.

III. ASS-BD DATASET FOR APPEARANCE TRANSPORTABILITY INSPECTION

A. PROPERTIES FOR SELF-SERVICE BAG DROP DATA

When people get close to the self-service bag drop equipment, the appearance transportability detection system is activated, which monitors the passenger's entire operation using the multi-category multi-object tracking algorithm proposed in this paper. And then the appearance transportability is estimated according to the detection results and the system prompts the passenger to proceed with the next step according to the obtained detection status. The type and number of baggage, whether tags are tied, and whether pallets are placed are obtained by the monitoring system which records the entire process of the passenger use the self-service bag drop. Based on this applicable scenario, our appearance transportability detection dataset should meet the following demands:

1) AIRPORT SCENE

The existing datasets are just individual pictures belong to target categories in life scenes, so it is necessary to collect the video data for self-service bag drop scenarios, as shown in Fig. 1(a).

2) SIMULATED LAB SCENE

Since there are relatively few samples of passengers' standard operations in the actually applicable airport scenarios, in order to increase the data that do not meet the appearance transportability, we also set up a simulation environment in the laboratory to record an extended dataset, as shown in Fig. 1(b).

3) COMPLEX BACKGROUND

When a large number of passengers line up to use the machine, it is very likely that many passengers with baggage helping each other are operating in the region of interest at the same time, which interfere with the detection. Therefore, it is necessary to record video data in the context of different complexity.

4) OBJECTS DIVERSITY

There may be backpacks and suitcases of different colors, shapes, sizes, and materials, pallets of different colors, and various positions of tied tags in the actual check-in scene. The dataset should try to meet the diversity of the objects.

B. DATA COLLECTION AND ANNOTATION

The ASS-BD dataset comprises 100 video sequences collected from multiple videos recorded by 231 volunteers, of which 67 were recorded at the airport and 33 were recorded in a laboratory simulation environment. The video is recorded by the USB interface camera mounted on the self-service bag drop device in a resolution of 960×720 with a frame rate of 25 fps, and the actual field of vision is shown in Fig. 3. The recorded video contains various ways of placing multiple baggage, as shown in Fig. 2, including both the "easy" samples and "hard" samples.



FIGURE 3. The actual camera field of vision.

The moving objects are labeled using the DarkLabel tool (<https://github.com/darkpgmr/DarkLabel>). The label information is saved in TXT, and every video clip corresponds to a TXT. The bounding boxes are described with (frame, id, cx, cy, w, h, label), where (cx, cy) and (w, h) are the center coordinates, width and height of the bounding box, respectively. The object class is represented with "label", and id represents the identity (e.g., "backpack1", "backpack2", "suitcase2"). The "frame" represents the serial number in the video sequences. About 66371 frames are recorded in the ASS-BD dataset in total, of which 65791 frames have annotated information. The average, shortest and longest lengths in 100 video sequences are 663.71, 4, and 4044, respectively. These objects are categorized into four classes: backpack, suitcase, tag, and pallet. Moreover, their number of statistics are shown in Table 2. The size of each frame is about 100 KB-200 KB, the total size of the dataset is 10 GB. The dataset is stored in the network disk, please contact the correspond-

TABLE 2. Summary of ASS-BD dataset.

Airport	Lab	volunteer	backpack	suitcase	tag	pallet
67 clips	33 clips	231	107	111	158	38

ing author (code) to download the dataset without hesitation (https://pan.baidu.com/s/1a5MSA8Ecy8ED90u1T_Jrww).

IV. SEQUENTIAL HIERARCHICAL SAMPLING MULTI-OBJECT TRACKER

In the method described in this section, deformable convolution DCNv2 [20] is used in many layers to replace traditional convolution and to improve the ability of the network to adapt to the geometric changes of the target, so we first review the DCNv2. Traditional 2D convolution is comprised of two steps: 1) use a grid R with weight information to sample on the input feature map x ; 2) calculate the weighted summation value of the grid R at the corresponding sampling position of the input feature map x . For example, if we consider a 2D convolution with a 3×3 kernel, and a dilation factor of 1, the grid R is defined as:

$$R = \{(-1, -1), (-1, 0), \dots, (0, 1), (1, 1)\}, \quad (1)$$

To calculate the output feature map y at the location p_0 on the input feature map x :

$$y(p_0) = \sum_{k=1}^9 w_k \cdot x(p_0 + p_k), \quad (2)$$

where w_k represents the weight of the grid R corresponding to the k position. According to the description in DCNv2, the improved convolution operation can be expressed as:

$$y(p_0) = \sum_{k=1}^9 w_k \cdot x(p_0 + p_k + \Delta p_k) \cdot \Delta m_k, \quad (3)$$

where Δp_k and Δm_k are the learnable offset and modulation scalar for the k -th location, respectively. This change can not only adjust offsets in perceiving input features, but also modulate the input feature amplitudes from different spatial locations. The modulation scalar Δm_k lies in the range $[0, 1]$, while Δp_k is a real number with unconstrained range [20].

In this section, we elaborate on the proposed architecture named sequential hierarchical sampling multi-object video tracker, which contains three components: the backbone based on the improved deep layer aggregation (DLA-34) [6] is used to extract image features as shown in Fig. 4, the sequential hierarchical sampling module is used to improve the adaptability of the overall network to video multi-target detection as shown in Fig. 6, four parallel heads are used to complete four tasks as shown in Fig. 7. DLA-34 aggregates multi-layer features in the network structure, and integrates semantic and spatial features to better improve inference of what and where, which can provide robust and accurate feature maps for subsequent operations with better

accuracy and fewer parameters. The network can adapt to the geometric changes of moving objects in the video due to the up-sampling operations with DCNv2 in Iterative Deep Aggregation. The sequential hierarchical sampling module is applied to fuse the features of the key areas in the previous frame with the features of the current frame, which can solve some difficult frames due to part occlusion, rare pose and motion blur.

A. CHALLENGES OF DETECTING APPEARANCE TRANSPORTABILITY

In our task, the detection and tracking of airline baggage, tags, pallets have some difficulties:

1) REAL-TIME

The self-service bag drop system is expected to interact with operators in real-time, so our algorithm needs to consider the trade-off between accuracy and speed.

2) THE PROBLEM OF DETECTION

Difficult frames for detecting such as part occlusion, rare pose, and motion blur will appear when the operator is placing baggage, tying baggage tags, and placing pallets.

3) THE PROBLEM OF TRACKING

Firstly, the bounding boxes of the current frame objects are obtained through the detection network. And then the re-ID features of objects are applied to correlate the bounding boxes between the current frame and the previous frame in the tracking stage, which needs to achieve fast.

B. BACKBONE

As shown in Fig. 4, the improved the DLA-34 is applied as the backbone to fuse multi-layer features and make a tradeoff between latency and accuracy. DLA-34 has more skip connections between low-level and high-level features to fuse pixel information of multiple scales, which is similar to the Feature Pyramid Network [21] for enhancing the network's ability to detect small targets.

As shown in Fig. 4, the network takes a frame of size $3 \times 1088 \times 608$ as input to train in DLA-34 backbone. Firstly, ① is obtained after two down-sampling operations, and then a specially designed HDA module is applied to fuse the low-level and high-level feature maps together, so that a rich combination of multiple feature layers can be learned by the model, and get the four feature layers with different channels and shapes are denoted as ②, ③, ④, and ⑤, respectively. These cross and merge stages aggregate different levels of representation and improve the network recognition and positioning abilities. As shown in Fig. 5, the generated feature maps ②, ③, ④ and ⑤ are used for up-sampling decode operations.

As shown by the yellow arrow line in Fig. 5, the iterative deep aggregation network used to increase feature map resolution symmetrically is different from the original DLA [22]. The convolution layers of all up-sampling modules are replaced by the 3×3 deformable convolution so that the

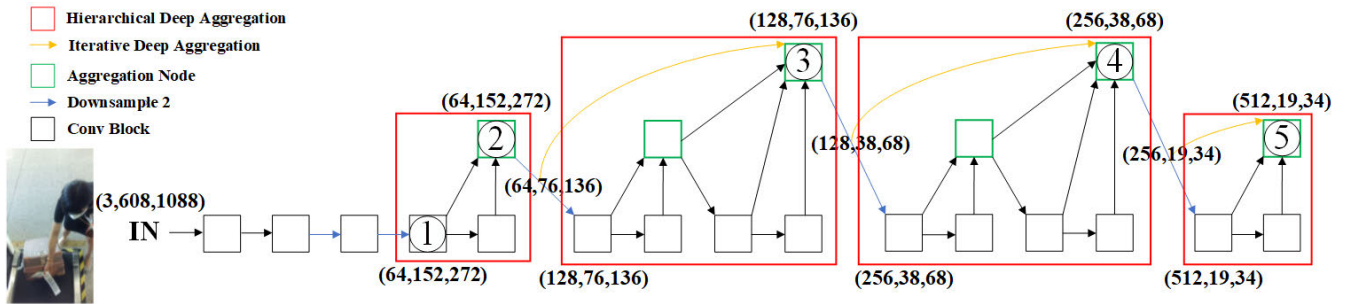


FIGURE 4. DLA-34 backbone. The feature maps, (C×W×H), are shown as the shape of their tensors. The Hierarchical Deep Aggregation(HDA) and Iterative Deep Aggregation(IDA) are designed to increase the depth of the network to obtain richer feature maps and better to extract the semantic and spatial information. ②, ③, ④ and ⑤ aggregated by different layers are applied to decode as shown in Fig. 5.

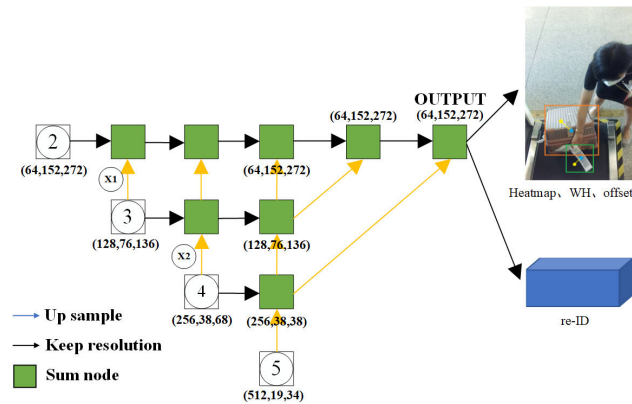


FIGURE 5. Decoder network. The feature maps, (C×W×H), are shown as the shape of their tensors. ②, ③, ④ and ⑤ are generated from DLA-34 backbone. X1 and X2 are generated by ③ and ④ respectively, are used to provide different scales feature maps in the sequential hierarchical sampling module shown in the Fig. 6.

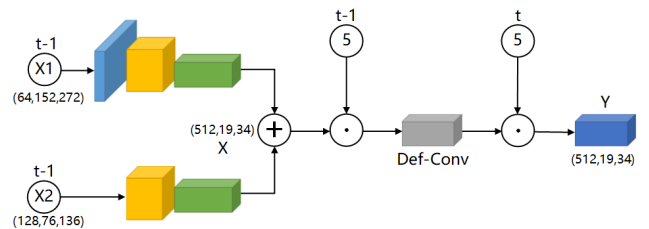


FIGURE 6. Sequential hierarchical sampling module. The feature maps, (C×W×H), are shown as the shape of their tensors, “+” denotes element-wise sum, “⊙” denotes Hadamard product.

C. SEQUENTIAL HIERARCHICAL SAMPLING MODULE

This module uses the detection features of the t-1 frame to guide the detection of the t frame, which can solve some difficult frames due to part occlusion, rare pose and motion blur. The word “sequential” means we fuse two frame features, and the word “hierarchical” means we fuse different scale features. The input of the sequential hierarchical sampling module comes from the X1, X2 and ⑤ feature maps at t-1 frame and the ⑤ feature map at the t frame respectively. The output Y of the module is used to replace the feature maps ⑤ in Fig. 5. The specific structure of the module is as follows.

Firstly, the frame at t-1 is sent to the DLA-34 backbone to extract feature maps ②, ③, ④, and ⑤ of different shapes and channels. As shown in Fig. 6, secondly, X1 and X2 generated at t-1 are respectively subjected to three and two convolution layers, batch normalization, and activation layers to unify the channels and sizes to (512, 19, 34). In order to enhance the network’s ability to detect objects of different scales, the unified channel and size feature maps are fused to generate X, which is fused with the feature map ⑤ at the t-1 frame through the Hadamard product. To mitigate the target motion blur and geometric deformation, the feature map after fusion is sampled by the deformable convolution layer to generate the guidance mask at t-1 frame, which is applied to fuse with the feature map ⑤ at frame t through the Hadamard product. At last, the generated feature map Y is used to replace the feature map ⑤ at t frame as the bottom feature map of the decoding network for recovering resolution. The Y of each video sequence first frame could not computed through the

model focus on the relevant area when the moving target is deformed.

As shown in Fig. 5, the feature maps ③ and ④ are subjected to a 3 × 3 deformable convolution operation for increasing the resolution of the feature map, and generate the X1 and X. According to the practice of FairMOT [13], the OUTPUT of the decoder network is taken as the input of two fair tasks that aim to detect and extract re-ID features.

Despite the Centernet detector preserves excellent performance in static images, it still lacks the ability of real-time robust detection when objects appear blurry or occluded in the key frame of video. STSN [23] has proposed a method of sampling from adjacent frames to guide the detection of the current frame, making the detection of occluded and motion-blurred individual image frames robust. As shown in Fig. 5, the encoder-decoder is designed to extract image features, but this architecture will make the network lose global information, while the sampling module we proposed could boost the representation of the baggage and tags region in the images. Therefore, we adopt the sequential hierarchical sampling module to enhance the feature map’s description of the interest target region, the output Y of the sequential hierarchical sampling module is used to replace the feature layer ⑤ in Fig. 5.

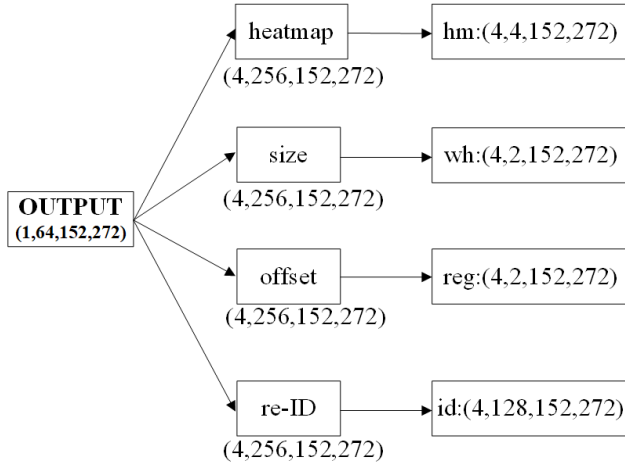


FIGURE 7. Four parallel heads for handling four tasks.

sequential hierarchical sampling module, so the ⑤ of the first frame in Fig. 5 is not replaced by Y.

As shown in Fig. 7, four parallel heads are attached to the OUTPUT to estimate heatmap, bounding box size, object center offset, and re-ID features, respectively. The composition of each head is: a 3×3 convolution with 64 input channels and 256 output channels, an activation layer, and a 1×1 convolution which generates the final targets.

The object category, center point, size, offset, and identity feature vectors are calculated through the model mentioned above, and then we match detection bounding boxes between video frames to complete the tracking part. First, we initialize the target trajectory through the estimation box of the first frame. Then we calculate the cosine distances through the re-ID features and the IOU of the inter-frame targets for using the Hungarian algorithm to solve the assignment problem [24]. We also use Kalman Filter to predict the position of the target in the next frame. When the predicted position is too far away from the current detection position, the corresponding cost is set to infinity to prevent matching errors.

D. TRAINING OBJECTIVE

In this section, we introduce the loss functions to train our proposed method. According to the practice in FairMOT [13], our training objective consists of four modules: The center point classification module, heatmap head, is responsible to predict the center point of the target category. The bounding box scale module, size head, is applied to estimate the height and width of the target bounding box. The center point offset module, offset head, is used to localize the target point more precisely. The re-ID features extraction module, re-ID head, is applied to extract the apparent features of the target.

According to the practice of training key point network proposed by Law and Deng [12], we predict the target category by training the target center point based on the heatmap. The center point P of each GT bounding box is marked as (x_c, y_c) . And the center point is \tilde{p}_R^P after down-sampling, where R is down-sampling factor 4 in literature [25]. All

ground truth key points in the input image are sprinkled onto a heatmap $Y_{xy} \in [0, 1]^{\frac{H}{R} \times \frac{W}{R} \times C}$ using a Gaussian kernel:

$$Y_{xy} = \sum_{i=1}^N \exp \left(-\frac{(x - \tilde{p}_x^i)^2 + (y - \tilde{p}_y^i)^2}{2\sigma_p^2} \right), \quad (4)$$

where σ_p^2 is an object size-adaptive standard deviation [12]. The loss function uses pixel-wise logistic regression function focal loss [28]:

$$L_{cls} = \frac{-1}{N} \sum_{xy} \begin{cases} (1 - \tilde{Y}_{xy})^\alpha \log(\tilde{Y}_{xy}) & Y_{xy} = 1; \\ (1 - \tilde{Y}_{xy})^\beta (\tilde{Y}_{xy})^\alpha \log(1 - \tilde{Y}_{xy}) & \text{otherwise,} \end{cases} \quad (5)$$

where N is the number of target center points in the input image, \tilde{Y}_{xy} is the estimated heatmap, and $\alpha = 2$, $\beta = 4$ in all our experiments following Law and Deng [12].

Because the resolution of the output will be reduced after the backbone network extracts the features, quantization errors will be introduced accordingly, and discretization error will be generated when predicting the position of the bounding box, so the center point offset head is added to mitigate the impact of down-sampling. Meantime, the size header of the bounding box is responsible for estimating the width and height of the bounding box. \tilde{S} and \tilde{O} are denoted as the size and offset of the bounding box, they are trained with L1 losses:

$$L_{\text{box}} = \sum_{i=1}^N \left(\|O^i - \tilde{O}^i\|_1 + \|S^i - \tilde{S}^i\|_1 \right), \quad (6)$$

The features extracted by the backbone are applied fairly for the detection task and re-ID task, and all targets with the same identity are treated as different categories for training, following the FairMOT [13]. We compute the re-ID loss as:

$$L_{\text{re-ID}} = - \sum_{i=1}^N \sum_{m=1}^m L^i(m) \log(p(m)), \quad (7)$$

where N represents the number of objects in the image, m is the number of classes. $p(m)$ is a class distribution vector mapped learnable re-ID feature vector. $L^i(m)$ is marked as the one-hot representation of the ground truth class label.

The overall objective loss function is:

$$L = \frac{1}{2} \left(\frac{1}{e^{\eta_1}} (L_{cls} + L_{\text{box}}) + \frac{1}{e^{\eta_2}} L_{\text{re-ID}} + \eta_1 + \eta_2 \right), \quad (8)$$

where η_1 and η_2 are learnable parameters that balance the detection and re-ID.

V. EXPERIMENTS

In this section, we conduct experiments to evaluate the performance of sequential hierarchical sampling multi-object tracker on the ASS-BD dataset. To increase the diversity of our model, we also designed an experiment using the supplemental baggage dataset, which is composed of backpacks and suitcases images selected from Google Open Image v4 [16].

TABLE 3. Summary of images in the ASS-BD.

Scene	Training set	Testing set
Airport	46753	13067
Laboratory	3940	2031

TABLE 4. Summary of bounding boxes in the ASS-BD training set.

Scene	Backpack	Suitcase	Tag	Pallet
Airport	12088	16279	55644	5747
Laboratory	6374	7159	1544	1748

TABLE 5. Summary of bounding boxes in the ASS-BD testing set.

Scene	Backpack	Suitcase	Tag	Pallet
Airport	2930	11466	13928	2691
Laboratory	3153	2379	488	1307

TABLE 6. Summary of images in supplemental baggage.

Class Number	Training set		Testing set	
	Backpack	Suitcase	Backpack	Suitcase
	717	408	84	68

Datasets: We propose the ASS-BD Airport Self-Service Bag Drop video dataset, which is suitable for the inspection task of appearance transportability in the applicable airport scenario. The statistical information of the images in the dataset is shown in Table 3, and the statistical information of the bounding boxes in the dataset is shown in Table 4 and Table 5. Our dataset is recorded in two locations, the applicable airport scenario and the laboratory simulation environment. There are 80 video sequences in the training set and 20 video sequences in the testing set. Data captured in the airport scene contains a total of 59820 frames, the training set contains 46,753 images with 89,758 bounding boxes, and the testing set contains 13067 images with a total of 31,015 bounding boxes. Data captured in the laboratory scene contains a total of 5971 frames, the training set contains 3940 images with 16825 bounding boxes, and the testing set contains 2031 images with a total of 7327 bounding boxes. In order to increase the diversity of suitcases and backpacks in the dataset, the images of backpacks and suitcases from Google Open Image v4 [16] are selected as a supplement to this dataset named supplemental baggage. The detailed summary of images in supplemental baggage is summarized in Table 6.

Implementation Details: Our module is based on the Pytorch framework with NVIDIA RTX2080 Ti. We adopt Adaptive Moment Estimation (ADAM) with an initial learning rate of $1e-04$ to train our model. The network takes a frame of size 1088×608 as input with the batch size is 4. We initialize our model by using the model parameters pre-trained on the MSCOCO [15]. We first train on the supplemental baggage dataset to increase the diversity of detection, and then the model we got is trained on the ASS-BD dataset through the network we proposed.

TABLE 7. Detection comparison on ASS-BD dataset.

Method	mAP(%)	Runtime(FPS)
Faster-rcnn (720×480) [10]	68.2	7
R-FCN (600×600) [30]	69.2	13
YOLOv3 (416×416) [11]	59.5	20
FGFA (600×600) [31]	74.8	1.2
Centernet (1088×608) [6]	70.6	49
OURS (1088×608)	73.1	30

TABLE 8. Detection comparison on supplemental baggage dataset.

Method	mAP(%)	Runtime(FPS)
Faster-rcnn (720×480) [10]	71.2	7
YOLOv3 (416×416) [11]	65.5	20
Centernet (1088×608) [6]	75.6	51
OURS (1088×608)	74.3	31

Evaluation Metric: We choose the mean Average Precision (mAP) to evaluate the performance of the detector setting the Intersection over Union (IoU) threshold to 0.5. We choose the multiple object tracking accuracy (MOTA) and the Identification F1 (IDF1) to evaluate the tracking performance of the model.

$$MOTA = 1 - \frac{(FN + FP + IDSW)}{GT} \in (-\infty, 1], \quad (9)$$

where GT is the number of ground truth boxes, FN is the number of false negatives in the whole video, FP is the number of false positives in the whole video, and $IDSW$ is the total number of ID switches.

$$IDF1 = \frac{2IDTP}{2IDTP + IDFP + IDFN}, \quad (10)$$

where the $IDTP$ is the sum of the weights of the edges selected as true positive ID matches (it can be seen as the percentage of detections correctly assigned in the whole video). $IDFN$ is the sum of weights from the selected false negative ID edges, and $IDFP$ is the sum of weights from the selected false positive ID edges [29].

In addition, frames per second (FPS) is used to evaluate the speed of the algorithm.

A. COMPARISON OF DETECTION PERFORMANCE

The generality of the convolutional neural network is one of the important indicators for evaluating the quality of the model, which reflects that the model has excellent performance on another similar dataset. When comparing the detection task with other detection methods only, the results of our model without using the apparent features of re-ID are shown in Table 7, which shows the specific method followed by the resolution of input image. It can be seen that in the case of higher-resolution input images, the sequential hierarchical sampling module proposed in this paper can increase the accuracy of model for video object detection. When dealing with the task of video object detection, compared with the classic anchor-based detection and video object detection methods, the method in this paper is more competitive in terms of speed and provides a good detection effect for the

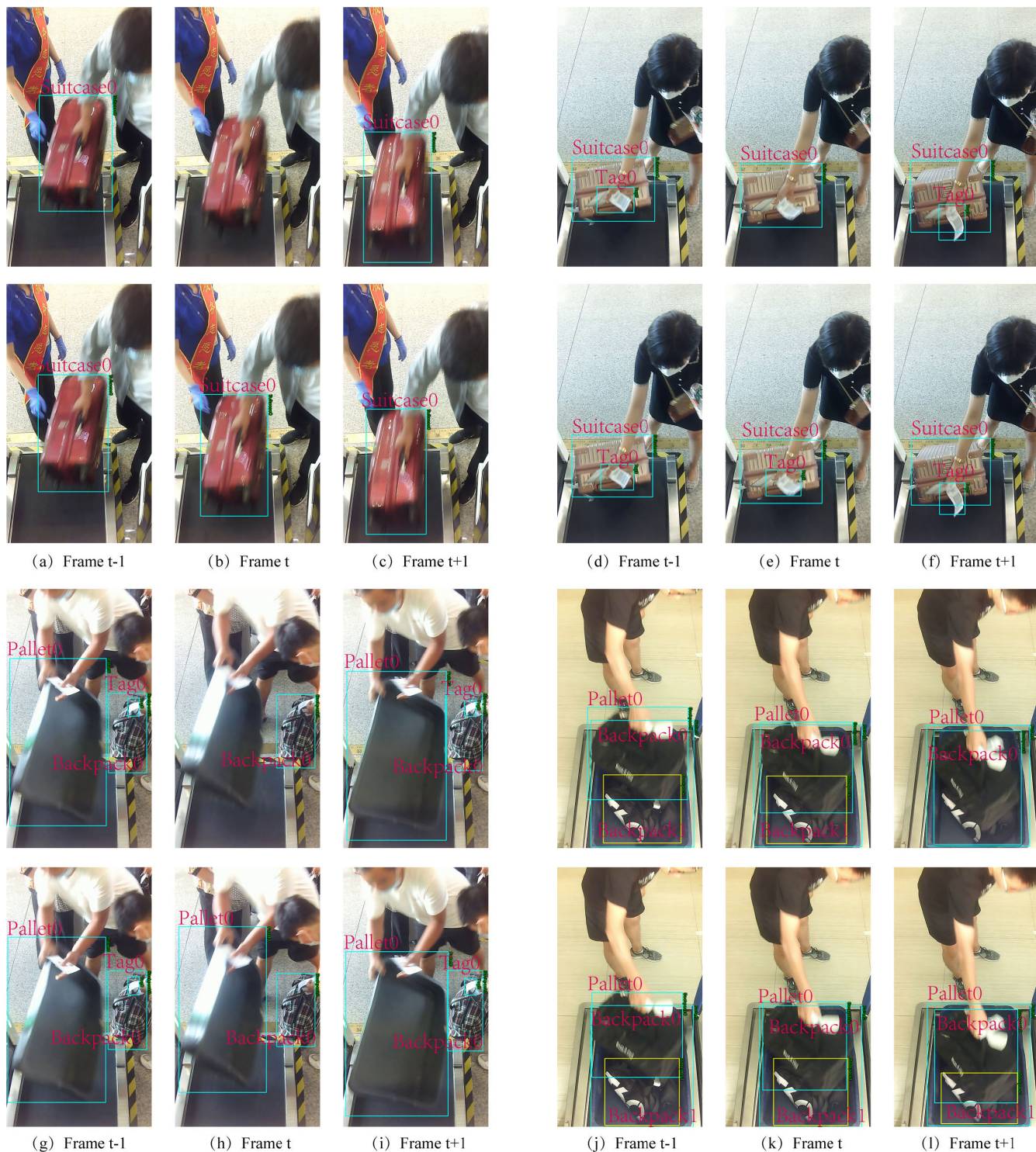


FIGURE 8. Examples of qualitative result. For each of the 6 images, the upper part is YOLOv3+DeepSORT result, the lower part is result of our method. Images (a), (b), (c) and (d), (e), (f) is a scenario where the motion object is blur, which leads to miss the suitcase and tag. Images (g), (h), (i) is a scenario where the rare pose appears, which leads to miss tag and pallet. Images (j), (k), (l) is a scenario where the part occlusion appears, there are two backpacks in the video. The same color of bounding boxes represents the same ID. The green label on the upper right corner is magnified and indicated by the pink label.

subsequent processing. In terms of the accuracy of video object detection, our method has a slight improvement over Centernet. Table 8 shows the comparison results of our model

and other detection methods on the supplemental baggage dataset. Because the Centernet is only designed for the detection task, but our method using the sampling module with

TABLE 9. Tracking comparison on ASS-BD dataset.

Dataset	Method	MOTA	IDF1	FPS
Training Set	YOLOv3+DeepSORT (416×416)[7]	65.7	63.3	20
	OURS(1088×608)	75.1	74.5	30
Testing set	YOLOv3+DeepSORT (416×416)[7]	60.4	61.2	18
	OURS(1088×608)	65.7	69.5	29

DCNv2 in this paper is designed for video object detection, our method is slower when the accuracy is similar to the Centernet based on the supplemental dataset.

B. COMPARISON OF TRACKING PERFORMANCE

We compared YOLOv3+DeepSORT [7] with our method on the ASS-BD dataset. As shown in Table 9, our method is significantly better than YOLOv3+DeepSORT in both MOTA and IDF1 tracking indicators, which indicates our method is more accurate in tracking tasks and can track the same target for a long time. We tested the speed of the model on the NVIDIA RTX2080 Ti GPU. It can be seen that our model runs at 29FPS under 1088 × 608 resolution, while YOLOv3+DeepSORT only has 18FPS at a resolution of 416 × 416. Our model may be used to applied scenario. We also conduct qualitative experiments to explain the sequential hierarchical sampling module we proposed. Some notable visual detection results are shown in Fig. 8.

For the detection task on the belt conveyor production flow line, we propose a video-based multi-category multi-target tracking algorithm to monitor the interest objects on the production flow line in real-time and provide accurate detection results for judging the working state at this time. Meanwhile, we provide a new dataset for the aviation baggage detection application of object tracking algorithms based on deep learning.

VI. CONCLUSION

Real-time visual appearance transportability detection is an important part of the interaction between self-service baggage drop equipment and people in the airport. In this paper, we propose a new dataset ASS-BD applied to the appearance transportability detection and a real-time sequential hierarchical sampling multi-object tracker. The object detection and multi-object tracking experiments we designed prove that the model has strong versatility when applied to the inspection task of appearance transportability, and it could solve some problems of motion blur, part occlusion, and geometric deformation. In the future, video datasets containing more baggage categories, volunteers will be collected. The proposed framework's running rate is 29 fps on the ASS-BD, and the model will be applied to NVIDIA JETSON TX2 in the future.

REFERENCES

- [1] Accessed: Mar. 2021. [Online]. Available: <https://www.airport-technology.com/contractors/baggage/bagdrop/>
- [2] Accessed: Mar. 2021. [Online]. Available: <https://www.airport-suppliers.com/supplier/icm-airport-technics/>
- [3] Accessed: Mar. 2021. [Online]. Available: <https://www.checkitxpress.com/>
- [4] Q. Gao, D. Yin, Q. Luo, and J. Liu, "Minimum elastic bounding box algorithm for dimension detection of 3D objects: A case of airline baggage measurement," *IET Image Process.*, vol. 12, no. 8, pp. 1313–1321, Aug. 2018.
- [5] D. Yin, Q. Gao, and Q. Luo, "Automatic airline baggage counting using 3D image segmentation," in *Proc. 2nd Int. Workshop Pattern Recognit.*, Jun. 2017.
- [6] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," 2019, *arXiv:1904.07850*. [Online]. Available: <http://arxiv.org/abs/1904.07850>
- [7] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 3645–3649.
- [8] S. Chen, Y. Wang, L. Huang, R. Ge, Y. Hu, Z. Ding, and J. Liao, "2nd place solution for Waymo open dataset challenge–2D object detection," 2020, *arXiv:2006.15507*. [Online]. Available: <http://arxiv.org/abs/2006.15507>
- [9] H.-M. Hsu, Y. Wang, and J.-N. Hwang, "Traffic-aware multi-camera tracking of vehicles based on ReID and camera link model," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 964–972.
- [10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
- [11] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: <http://arxiv.org/abs/1804.02767>
- [12] H. Law and J. Deng, "Corners: Detecting objects as paired keypoints," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 765–781.
- [13] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "FairMOT: On the fairness of detection and re-identification in multiple object tracking," 2020, *arXiv:2004.01888*. [Online]. Available: <http://arxiv.org/abs/2004.01888>
- [14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
- [15] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 740–755.
- [16] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Mallocci, A. Kolesnikov, T. Duerig, and V. Ferrari, "The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale," 2018, *arXiv:1811.00982*. [Online]. Available: <http://arxiv.org/abs/1811.00982>
- [17] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "MOT16: A benchmark for multi-object tracking," 2016, *arXiv:1603.00831*. [Online]. Available: <http://arxiv.org/abs/1603.00831>
- [18] S. Lyu, M. C. Chang, D. Du, W. Li, and Y. Wei, "UA-DETRAC 2018: Report of AVSS2018 & IWT4S challenge on advanced traffic monitoring," in *Proc. 15th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Nov. 2018, pp. 1–6.
- [19] P. Zhu, L. Wen, D. Du, X. Bian, Q. Hu, and H. Ling, "Vision meets drones: Past, present and future," 2020, *arXiv:2001.06303*. [Online]. Available: <http://arxiv.org/abs/2001.06303>
- [20] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable ConvNets v2: More deformable, better results," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9308–9316.
- [21] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.
- [22] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2403–2412.
- [23] G. Bertasius, L. Torresani, and J. Shi, "Object detection in video with spatiotemporal sampling networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 331–346.
- [24] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Res. Logistics Quart.*, vol. 2, nos. 1–2, pp. 83–97, Mar. 1955.
- [25] Z. Cao, T. Simon, S. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7291–7299.
- [26] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 483–499.

- [27] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy, "Towards accurate multi-person pose estimation in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4903–4911.
- [28] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.
- [29] G. Ciaparrone, F. Luque Sánchez, S. Tabik, L. Troiano, R. Tagliaferri, and F. Herrera, "Deep learning in video multi-object tracking: A survey," *Neurocomputing*, vol. 381, pp. 61–88, Mar. 2020.
- [30] T. Chin, R. Ding, and D. Marculescu, "AdaScale: Towards real-time video object detection using adaptive scaling," 2019, *arXiv:1902.02910*. [Online]. Available: <https://arxiv.org/abs/1902.02910>
- [31] X. Zhu, Y. Wang, J. Dai, L. Yuan, and Y. Wei, "Flow-guided feature aggregation for video object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 408–417.
- [32] Accessed: Mar. 2021. [Online]. Available: <https://github.com/darkpgmr/DarkLabel/>



PEIWEN LIANG received the B.S. degree from Shanghai Dianji University, in 2018. He is currently pursuing the M.E. degree in control science and engineering with the Civil Aviation University of China. His current research interests include computer vision and multi-object tracking.

...



QINGJI GAO received the Ph.D. degree in computer application technology from the Harbin Institute of Technology, in 2006. He is currently a Professor with the Intelligent Robotics and Autonomous System Laboratory, Civil Aviation University of China. His major research interests include intelligent robot control and computer vision.