electronics

MDPI

Article

A Robust Quadruplet and Faster Region-Based CNN for UAV Video-Based Multiple Object Tracking in Crowded Environment

Happiness Ugochi Dike and Yimin Zhou *

Center for Intelligent Bionic, Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China; happiness@siat.ac.cn

* Correspondence: ym.zhou@siat.ac.cn; Tel.: +86-755-8639-2659

Abstract: Multiple object tracking (MOT) from unmanned aerial vehicle (UAV) videos has faced several challenges such as motion capture and appearance, clustering, object variation, high altitudes, and abrupt motion. Consequently, the volume of objects captured by the UAV is usually quite small, and the target object appearance information is not always reliable. To solve these issues, a new technique is presented to track objects based on a deep learning technique that attains state-of-the-art performance on standard datasets, such as Stanford Drone and Unmanned Aerial Vehicle Benchmark: Object Detection and Tracking (UAVDT) datasets. The proposed faster RCNN (region-based convolutional neural network) framework was enhanced by integrating a series of activities, including the proper calibration of key parameters, multi-scale training, hard negative mining, and feature collection to improve the region-based CNN baseline. Furthermore, a deep quadruplet network (DQN) was applied to track the movement of the captured objects from the crowded environment, and it was modelled to utilize new quadruplet loss function in order to study the feature space. A deep 6 Rectified linear units (ReLU) convolution was used in the faster RCNN to mine spatial–spectral features. The experimental results on the standard datasets demonstrated a high performance accuracy. Thus, the proposed method can be used to detect multiple objects and track their trajectories with a high accuracy.

Keywords: quadruplet network; deep convolutional neural network; visual detection and tracking; unmanned aerial vehicle

Citation: Dike, H.U.; Zhou, Y. A Robust Quadruplet and Faster Region-Based CNN for UAV Video-Based Multiple Object Tracking in Crowded Environment. *Electronics* **2021**, *10*, 795. https://doi.org/10.3390/electronics10070795

Academic Editor: Francesco Beritelli

Received: 11 January 2021
Accepted: 16 March 2021
Published: 27 March 2021

1. Introduction

Object tracking [1] is a significant task in computer vision applications based on unmanned aerial vehicles (UAVs). Compared to single object tracking (SOT), the task of multiple object tracking (MOT) has to develop the trajectories of all the objects in a precise scene of video surveillance [1,2]. Online MOT two-dimensional space is a complex task when there are similar objects [3]. In computer vision, the MOT task mainly includes action recognition [4], behavior analysis [5], and pose estimation [6]. With the enhancement of object detection methods such as the single-shot detector (SSD), the faster region-based convolutional neural network (RCNN), and the deformable part model (DPM), tracking by a prediction framework possesses high performances for MOT because prediction can provide object location and object trajectories. Assignment could be inferred by various problems such as context [7], shape [8], motion and appearance [9], out-of-plane rotations [10], and background clutter [11]. The above problems are more difficult in MOT than in SOT [11].

In recent years, UAVs have been increasingly used in surveillance and traffic monitoring due to their ability to cover great distances while being remotely controlled to travel at a predefined speed [12], combined with the computer vision technology in security monitoring [13,14]. Meanwhile, UAV videos are different than conventional videos. Generally,

objects are small and difficult to differentiate due to their appearance. In such a situation, motion is an important piece of data for linking objects. However, due to high movement and altitude of UAVs in videos, the motion of an object is difficult to discriminate, thus resulting in high challenge for MOT, and the motion of UAVs causes image instability and platform motion. In this study, object refers to a car or human. In UAVs, the video motion of an object can be divided into two types of motion models: object motion and view modifications or the appearance feature of the UAV.

Many methods have been proposed to solve these problems, including a simple tracking-by-detection method such as the intersection over union (IoU) tracker that can only achieve a good result when objects are in good viewpoints [15]. A deep association metric that considers the deep appearance feature and information motion of an object while matching was proposed in [16]. The authors of [17] handled detection issues and improved the appearance feature through the collection of samples from an output model. Based on deep learning, image processing algorithms have made it easier to achieve auto-navigation in commercial UAVs. Additionally, deep learning researchers have developed fast and end-to-end trackers such as Cascade RCNN [18], FlowNet [19], DaSiameseRPN (region proposal network) [20], simple online and real-time tracking (SORT) [21] and DeepSORT [22] for MOT. However, all these trackers have at least one limitation, especially in the larger detection of pedestrian or car and at a higher frame rate per seconds (FPS).

To predict the motion of an individual object from the crowded objects, a faster RCNN was designed in this study with numerous important features such as multi-scale training, negative mining, and concatenation to improve the region-based CNN baseline. Similarly for the tracking motion of an individual object, a deep quadruplet network (DQN) was introduced to track the predicted objects from the crowd of objects, and the accuracy of tracking can be improved by using a quadruplet loss function and a deep CNN. We examined our tracking model with the existing algorithms based on motion prediction with certain indexes such as the IDF1-identity F1 score, as well as datasets such as the Unmanned Aerial Vehicle Benchmark: Object Detection and Tracking (UAVDT) and Stanford Drone datasets for the state-of-the-art comparisons. The major contributions of this paper are summarized as:

- A quadruplet network is proposed by employing a faster region-based convolutional neural network with novel generated image G-16 (VGG-16) network to train the fine-tuned data for the prediction of individual motion from a crowded object.
- A newly optimized background learning method that uses boundary-based quadruplet with deep networks is introduced to track prediction objects from crowded environments.
- We applied the proposed quadruplet method in some of the recent state-of-the-art comparison benchmarks, such as the UAVDT and Stanford Drone datasets. The re results shows outstanding performance.

2. Related Work

Based on the architecture of deep learning, many methods have been proposed to solve MOT problems. In [23], a novel DQN was introduced to analyze the potential connections between training samples, with the intention to attain a powerful representation. This was modelled by a shared network with four branches that acquired numerous sample, tuples as inputs which were linked by new loss function containing triplet loss and pair loss. Related to similar metrics, similar and dissimilar samples were selected as the positive and negative inputs, respectively, of triplet loss from each tuple. Additionally, a weight layer was presented to automatically choose appropriate combinations of weight in order to eliminate conflict among pair loss and triplet. The authors of [24] connected some benefits of deep learning with data associated with tracking. They introduced a DAN (deep affinity network) to study the compact features of pre-predicted objects at various stages of abstraction and conducted comprehensive pairing permutations of those features in any two frames to gather object affinities. The DAN accounted for numerous objects disappearing and appearing among video frames in MOT.

A unite re-identification (RE-ID) model and detector within an end-to-end network was proposed in [25] by adding an extra track branch for monitoring a faster RCNN architecture. This network could train the entire model with multi-loss. The RE-ID design in deep simple online real-time tracking allowed for the mining of the feature map from the predicted object images and the region of interest (RoI) feature vector in the faster RCNN baseline to minimize calculations. The triplet loss function was utilized to optimize and track the branch, while the neighboring frame was concatenated in video to build the dataset. The authors of [26] proposed deep quadruplet appearance learning (DQAL), where the quadruplet network was considered as an input and each quadruplet consisted of anchor, negative, positive, and similar vehicles with different IDs. Then a quadruplet network with softmax loss and quadruplet loss was created to study more discriminative features for vehicle Re-ID. Deep learning was utilized to resolve the issue of pedestrian prediction from drone images [27]; 1500 images were collected by an SJRC S30W drone at various places with varying weather conditions, and different time spans were applied for pedestrian detection with high performance.

The authors of [28] presented a UAV dataset that addressed numerous challenges such as large camera motion, small object, high density, and complex scenes to predict and monitor them. These challenges inspired researchers to state benchmarks for the three fundamental tasks of MOT, SOT, and object prediction on a UAV dataset. Additionally, a new technique based on CMSN (context-aware multi-task Siamese network) was introduced to resolve issues in UAV videos by interceding for the degree of consistency among contexts and objects that were utilized for MOT and SOT. In [29], a feature selection based on new intuitionistic fuzzy clustering technique was developed for MOT. The adaptive selection from the visual objects was comprised of several object features using the neighborhood rough set that could measure the distance similarity among observations and objects, proving to be efficient in robust environment. The authors of [30] suggested a spatial–temporal scheme to study MOT in different scenarios. The location and insertion of candidates were addressed by a novel motion model and semantic-feature spatial attention mechanism. Then, the target occlusion estimated by the online-learned CNN of specific target and appearance model were classified by adaptation in [31]. The authors proposed a target re-detection model over the region that could adaptively learn to perform and handle the existing scale estimation challenge. The noisy feature and channel redundancy representation in convolutional features were overcome via the correlation filter learning of channel regularization.

The target detection in a video sequence is the most essential stage in aerial surveillance for subsequent processing, while the performance of surveillance system depends on the detection of motion of any object in video. Therefore, understanding the moving patterns of objects and persons is required to uncover suspicious events. In the literature, various techniques for moving object tracking and detection in video have been proposed. The anomaly detection methods can be divided into two types: machine learning-based and pattern recognition-based methods. The disadvantages of existing methods are that they have a minimum accuracy, are time-consuming, and have difficulties in the processing of noisy data.

3. Materials and Methods

The proposed methodology was based on a faster RCNN of a deep learning framework, with its structure summary divided into two parts: RoI represents the position of the objects in the frame, and the fast RCNN network categorizes the region of image to objects and filters the boundaries of the region. These two parts share the general parameter of a convolutional layer that is utilized for feature extraction and for permitting the architecture to include object prediction tasks at a competitive speed. Here, a faster RCNN was designed to predict objects with a high accuracy and recall rate. The trained proposed methodology is depicted in Figure 1.

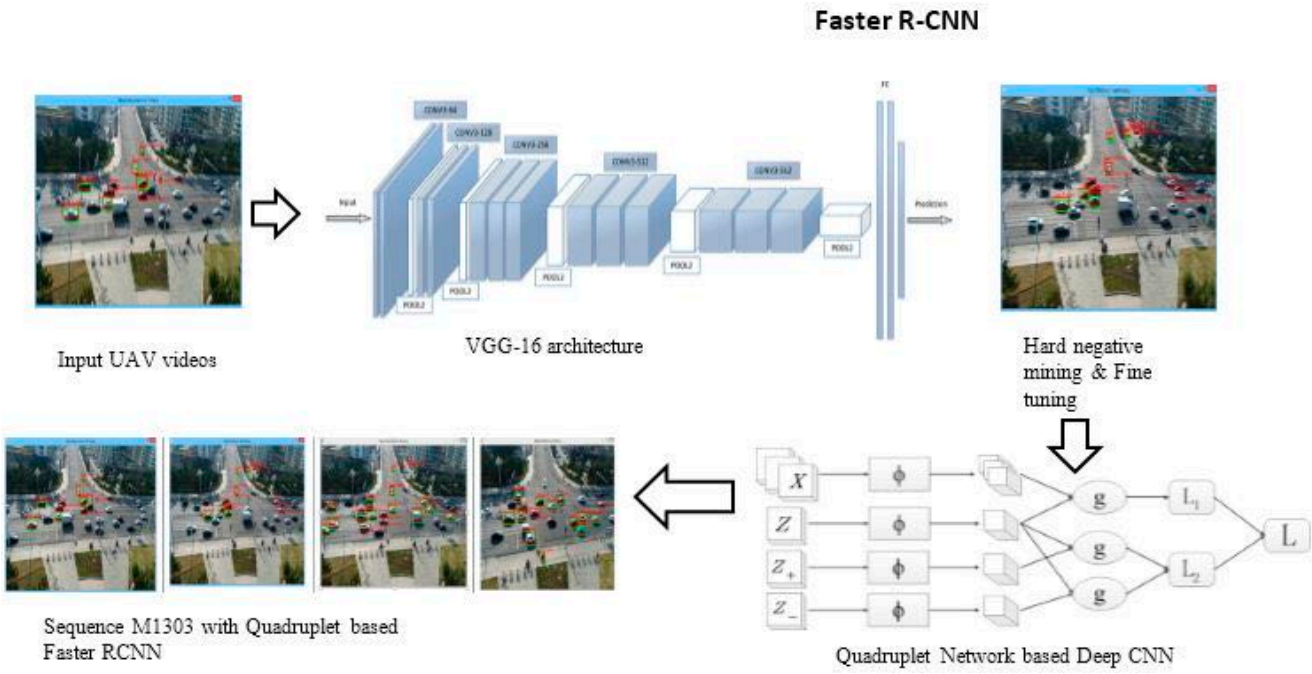


Figure 1. Quadruplet network-based faster region-based convolutional neural network (RCNN) architecture. The unmanned aerial vehicle (UAV) videos are fed into the novel generated image G-16 (VGG-16) network model. At this stage, the hard negative mining were fine-tuned and moved to the quadruplet network based Deep CNN. Finally, the resulting output is shown in sequence M1303 benchmark.

Primarily, a faster RCNN was trained with the Stanford Drone and UAVDT datasets. This dataset was further used to test the pre-trained model to enable it find the hard negatives and use them as the given input to the network in the 2nd step of the training process. By training with this kind of hard negative sample, the model developed fewer false positives. Moreover, this model was fine-tuned on the dataset. During the fine-tuning, a multi-scale training process was used and implemented for feature concatenation to boost the performance of the proposed model. For the training process, we utilized a faster RCNN to enhance the performance of object prediction. Finally, the resulting predicted bounding boxes were converted to ellipses as regions of objects. Below is the summary of our methodology.

3.1. Feature Concatenation

Normally, in a conventional fast RCNN network, RoI pooling is done in the final feature map layer to develop the features that are examined by the classification part of the network. The inspired model made the classification process to use features that were measured from the risk priority number and also preserved unwanted calculations. However, such a technique is not optimal at all times, and, in some cases, it could eliminate certain significant features. Thus, in the proposed work, to capture a finely grained parts of the RoI, it was required to enhance the RoI pooling using the integrated feature maps of numerous convolutional layers with high and low level features.

Therefore, we concatenated the pooling result of numerous convolutional feature maps to develop the final pooling features for an accurate prediction tasks. This means that we used few intermediate results with a risk priority number final feature map by integrating them to provide the pooling features. In particular, the features that were from many lower-level convolution layers were mainly L2-normalized and RoI-pooled. Then the resultant features were combined to obtain the matching feature scale. Afterwards, a 1×1 convolution was used to match the channels of the original network.

3.2. Hard Negative Mining

Hard negative mining is an efficient technique to boost deep learning performance, mainly for object prediction [32]. The concept of this technique implies that a hard negative is the region where a network fails to make right prediction. Therefore, a hard negatives are given as inputs to the network as reinforcements to enhance the trained model with a minimized number of false positives and the best classification performance.

In our design, hard negatives were gathered from the pre-trained model of the training process. Then, a region was considered as a hard negative if its IOU (insertion over union) and over a region ground truth was less than 0.5. During the training process, the hard negatives were explicitly added to the RoI to tune the model and balance the ratio of the background and foreground to 1:3.

3.3. Model Pre-Training

To adjust the faster RCNN for object prediction, we chose to fine-tune the pre-trained model from the ImageNet dataset. Our model was pre-trained on the datasets with numerous complex instances, which might have affected the convergence of the training process. To overcome these difficulties, certain training data had to be removed. Additionally, for a pre-training on this dataset, hard negative mining was a vital way to minimize false positives.

3.4. Multi-Scale Training

A faster RCNN was used for a fixed scale in the whole training of images. Then, the image was resized to a random scale, and a predictor is utilized to study the features over a broad range of sizes, thus enhancing its performance of scale invariance. Here, among the three scales, one scale was assigned to an image before it was inputted to network.

3.5. Number of Anchors

The various key hyper-parameters in the faster RCNN had to be turned, and the number of anchors was crucial; each was found within the region proposal network, where the conventional faster RCNN utilized nine anchors, which made it fail to recall smaller objects. For object prediction, small objects are the most common, mainly for unconstrained prediction. Hence, instead of utilizing the default setting, a size of $64 * 64$ was added to increase the number of anchors to 12.

3.6. Boundary-Based Quadruplet Network

The DQN is trained to study the feature space. Testing data were moved from learning feature space to mine features. Classification was achieved with the help of the CNN classifier: $\epsilon(w) = \frac{1}{2}\|y - \sum_{i=1}^{I} B\ x^i * c^i\|_2{}^2 - \sum_{i=1}^{I} \|\ \mathrm{pmc^i}\ \|_2{}^2$, where pm denotes penalization matrix and c^i represents the dth channel of the correlation filter.

3.7. Deep Network

The deep CNN block contained 6 convolutional layers. "Conv" is the convolutional operation with $3 \times 3 \times 3$ kernels, while "2 Conv" the denotes convolutional layer with 2 kernels. Normally, a CNN block of six layers consists of 6 connections. However, as the network becomes deeper, issues with the normal CNN arise, i.e., the features in the input vanish once it travels through several layers until it comes to the end. Thus, we preferred to use a deep CNN with 6 layer connections, where 3 connections were between the 1st, 3rd, 4th, 5th, and 6th layers, and 2 connections were between the 2nd and 6th layers. Figure 2 illustrates the connection points in the dense CNN, and $\oplus$ indicates the sum of all imported connected lines. The dense layer resolved issues that were related to information vanishing while passing over multiple layers, and it also utilized the features mined by the entire layers.

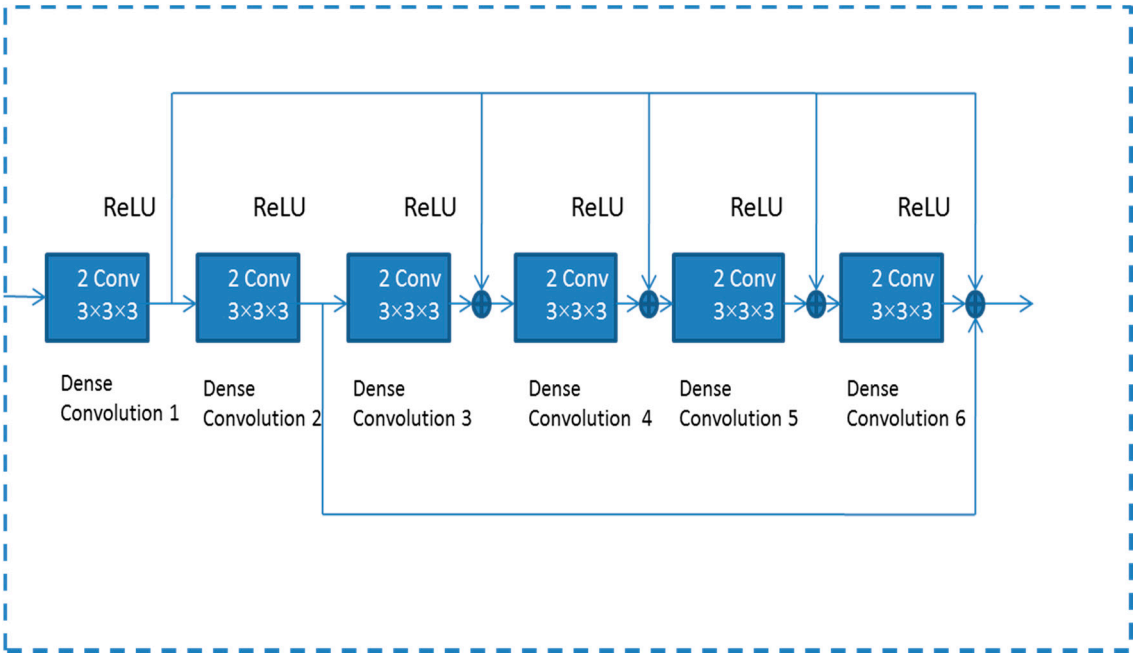


Figure 2. Deep CNN architecture.

3.8. Quadruplet Learning

Metric learning means to transfer the input data to new feature space $S^{(C)}$ (e_θ: $S^{(E)} \rightarrow S^{(C)}$ from the original space, where E denotes the dimension of original the space, C denotes the dimension the of new space, and θ is the learning parameter. In the new feature space $S^{(C)}$, the samples from the same class were assumed to be nearer than those from the diverse classes so that the classification could be completed in $S^{(C)}$ with the NN. A network such as a quadruplet network [33] could be created.

3.9. Quadruplet Loss (QL)

Here, the quadruplet loss (QL) included 4 various samples: $y_p^{(i)}$, $y_q^{(i)}$, $y_{n1}^{(i)}$, and $y_{n2}^{(i)}$, where $y_p^{(i)}$ and $y_q^{(i)}$ are samples from the same class and $y_{n1}^{(i)}$ and $y_{n2}^{(i)}$ are sample of another class. All samples were moved to the feature space by e_θ: $S^E \rightarrow S^C$. QL is given by

$$\mathrm{LF}_{\mathrm{quad}} = \frac{1}{\mathrm{NUM}_{\mathrm{quad}}} \sum_{i=1}^{\mathrm{NUM}_{\mathrm{quad}}} \left(ed\left(y_p^{(i)}, y_q^{(i)}\right) - ed\left(y_p^{(i)}, y_{n1}^{(i)}\right) + \epsilon\right)_+ \left(ed\left(y_p^{(i)}, y_q^{(i)}\right) - ed\left(y_{n1}^{(i)}, y_{n2}^{(i)}\right) + \Omega\right)_+ \quad (1)$$

where Ω and ϵ are margins of 2 terms, and $\mathrm{NUM}_{\mathrm{quad}}$ denotes the number of quadruplets. The 2nd term of the QL limits the intra-class distance and causes it to be smaller than the inter-class distances. However, the loss function in Equation (1) performed poorly since the quadruplet pairs and the number of quadruplet grew faster while the dataset got broader. In certain cases, all the samples were not sufficient to train the network, which led to poor performance [33]. To overcome all the issue, a new QL function was designed:

$$\mathrm{LF}_{\mathrm{nquad}} = \frac{1}{\mathrm{NUM}_{\mathrm{nquad}}} \sum_{i=1}^{\mathrm{NUM}_{\mathrm{nquad}}} \left(ed\left(y_p^{(i)}, y_q^{(i)}\right) - ed\left(y_m^{(i)}, y_n^{(i)}\right) + \epsilon\right)_+ \quad (2)$$

where $y_q^{(i)}$ denotes the farthest sample to the reference $y_p^{(i)}$ in the same class, $y_m^{(i)}$ and $y_n^{(i)}$ are the nearby negative pairs in the entire batch, $\mathrm{NUM}_{\mathrm{nquad}}$ is the number of quadruplet in the new loss function, and ϵ is the value of the margin. Each sample in Equation (2) was moved by e_θ: $S^E \rightarrow S^C$. The pseudocode for the batch training with the proposed loss function is expressed in Algorithm 1, where TD = {td_1, td_2, td_3, … , td_s} is the training

dataset for this batch and s is the number of the labelled samples. As said in the algorithm, NUM_{nquad} = r and td_j, td_k indicate the sample in the dataset TD. Meanwhile, td_j and td_k indicate the pair of samples, C(td_j) is a class of label of sample td_j, and ∀ represents the learning rate. td_p, td_q, td_m, and td_n are the quadruplets before the deep network, while y_p, y_q, y_m, and y_n are the corresponding quadruplets after the deep network.

Algorithm 1. Quadruplet Network

1. Input: The training dataset for this batch TD = {td_1, td_2, td_3, . . . , td_s}
2. The initialized or updated learnable parameter θ.
3. For all pairs of samples (td_j, td_k) in TD, DO
4. Calculate the Euclidean distance ed (f_θ (td_j), f_θ (td_k))
5. End For
6. Set the loss LF_{quad} = 0
7. Set (m, n) = argmined(j,k) ($f_\theta(td_j)$, $f_\theta(td_k)$), under the condition $C(td_j) \neq C(td_k)$
8. $y_m = f_\theta$ (td_m), $y_n = f_\theta$ (td_n). (Transfer td_m, td_n to y_m, y_n by the network f_θ: EG→EH)
9. For td_p in the dataset TD, DO
10. $y_p = f_\theta$ (td_p). (Transfer td_p to y_p by the network f_θ: EG→EH)
11. Set p = argmaxed(j) ($f_\theta(td_j)$, y_p), under the condition $CL(td_j) = C(td_p)$
12. $y_q = f_\theta$ (td_p). (Transfer td_p to y_q by the network f_θ: EG→EH)
13. Update: $LF_{quad} = LF_{quad} + \frac{1}{NUM_{nquad}}(d(y_p, y_q) - d(y_m, y_n) + \epsilon)+$
14. End For
15. Update: $\theta = \theta - \forall\Delta\theta\ LF_{nquad}$
16. Output: θ, LF_{nquad}

4. Results

The proposed tracking technique quadruplet-based faster RCNN was tested with existing algorithms, such as Bayesian multi-object tracking (BMOT) [34], intersection-over-union tracker (IOUT) [15], global optimal greedy (GOG) algorithm [35], continuous energy minimization (CEM) [36], social long short term memory (SLSTM) [37], simple online and real-time tracking (SORT) [21], relative long short term memory (RLSTM) [38], and relative motion online tracking (RMOT) [39]. To examine the performance of the MOT techniques, we utilized numerous metrics such as identification precision (IDP), identification recall (IDR), and F1 score, which are together referred as IDF1. IDF1 is stated as

$$\text{IDF1} = \frac{2\text{IDTP}}{2\text{IDPP} + \text{IDP} + \text{IDN}}$$

where IDPP represents the ID of the true positives, IDP represents the ID of the false positive, and IDN represents the ID of the false negative. Additionally, the accuracy, precision of MOT, mostly tracked target, mostly lost target, count of false positives, count of false negatives, count of ID switches, and count of time trajectory are fragmented.

4.1. Datasets

The proposed method was executed on the public Stanford Drone and UAVDT datasets [7,33]. The UAVDT dataset provided a sequence of 50 records of the traffic state from UAVs. There are four types of prediction results for MOT, faster RCNN [40], reverse connection with objectiveness prior network (RON) [41], Single Shot Multi Box Detector (SSD) [42], and region-based fully convolutional network (R-FCN) [42]. The Stanford Drone dataset consists of sequences of 60 videos that were recorded from a high altitude looking down by a flying UAV on a campus scene. The gathered videos consisted of several kinds of objects, as seen in Figure 3.

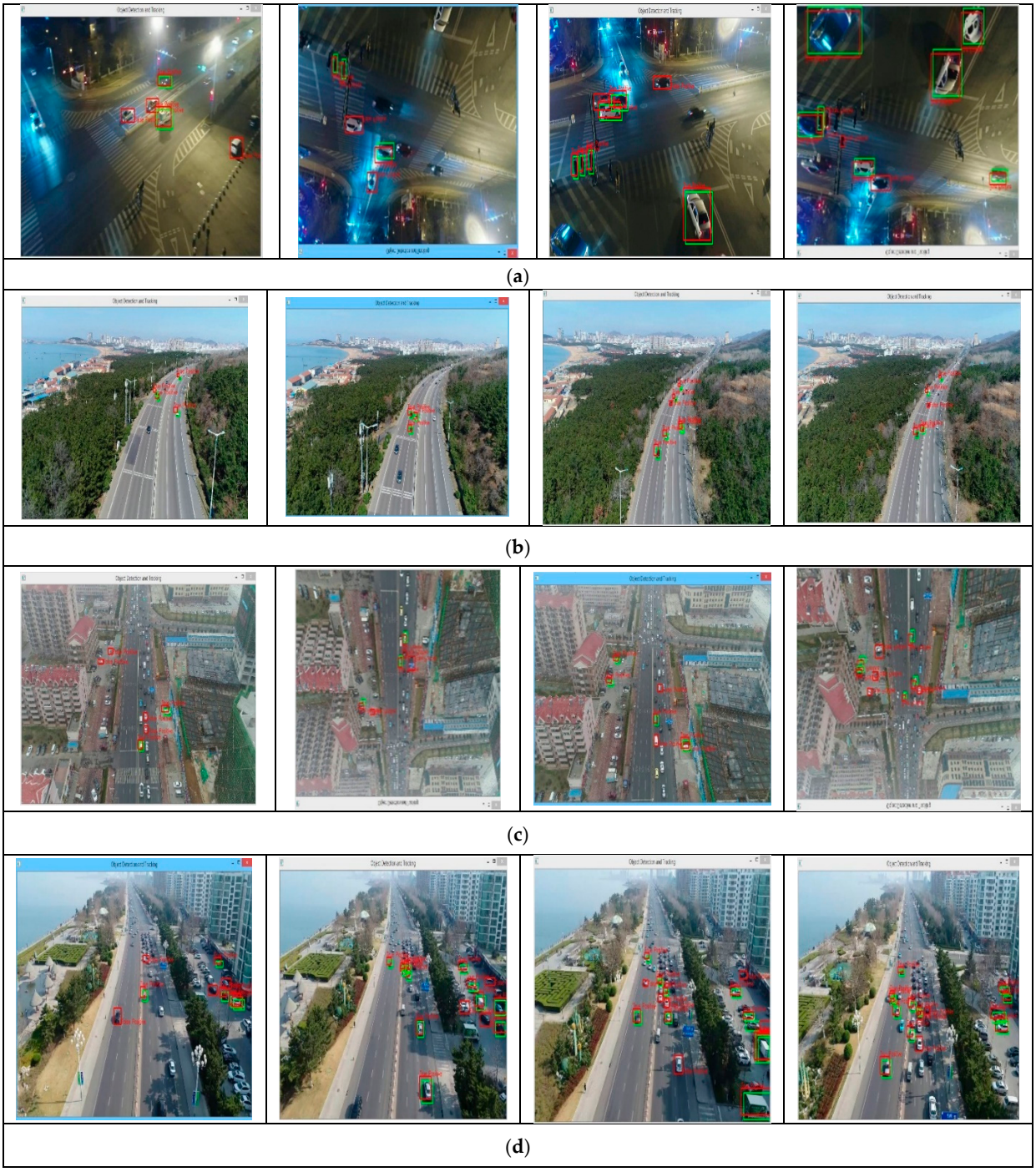
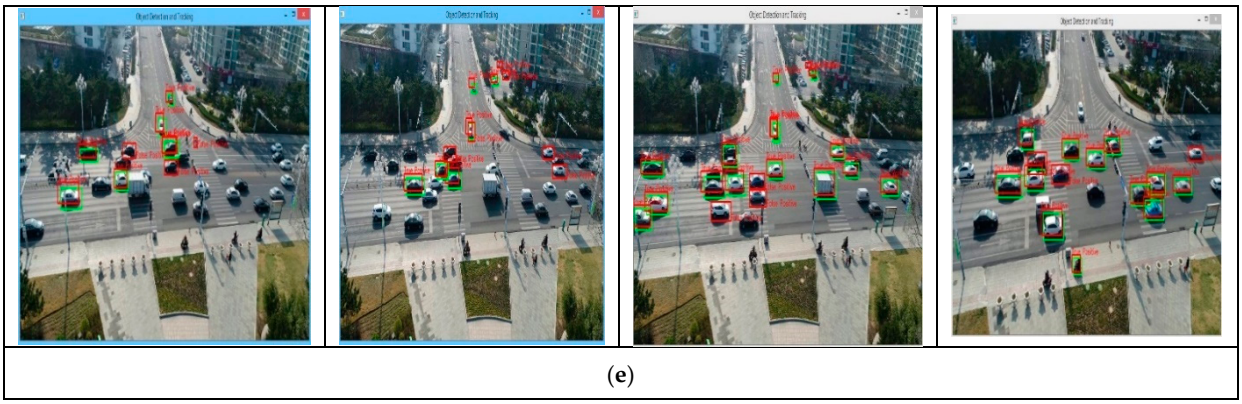


Figure 3. *Cont.*



(**e**)

Figure 3. Five sequences of videos in different benchmarks. (**a**) M0204 sequence with quadruplet-based faster RCNN, (**b**) M0301 sequence with quadruplet-based faster RCNN, (**c**) M0704 sequence with quadruplet-based faster RCNN, (**d**) M1301 sequence with quadruplet-based faster RCNN, and (**e**) M1303 sequence with quadruplet-based faster RCNN.

4.2. Evaluation

We utilized four types of object prediction results as inputs to the quadruplet-based faster RCNN on the UAVDT datasets. It was noticed that our quadruplet-based faster RCNN attained good results for the basic metric of IDP and IDF1, while SORT attained an IDF1 of 37.1 and SDD achieved an IDF1 of 54.6. Compared to other prediction technique, our proposed technique had great improvements on most of the metrics and could also handle most difficult trajectories. This means that with the proposed approach, the prediction of motion of object trajectories was stronger. The highest value of the IDF1 score illustrates that the proposed technique was more effective.

The UAVDT dataset consists of four types of attributes (camera view, weather condition, flying altitude, and time attribute such as duration). The proposed approach was examined for those four attributes. Regarding the camera view attribute, our technique performed very well for the inside, front, and bird views, as the images captured from all these views had more object information that helped in the object prediction and tracking. Our proposed technique attained good result because the scene information of the side and front views was needed for global motion analysis. Regarding the weather condition attribute, most of the techniques performed well in night and day but failed to perform well in fog, as the view was altered in the night and day. However, our proposed approach did well in all conditions. Regarding the flying altitude attribute, most other MOT technique were reduced with increasing altitude. However, our proposed approach performed very well at the medium and low altitudes, where there were drastic changes in view due to UAV movements, which caused wrong predictions for the other techniques. Regarding the duration attribute, the proposed approach performance is stable for short and long term with robustness of the proposed method.

We also executed our proposed approach on the Stanford Drone dataset. As it was difficult to predict objects in this type of dataset, the ground truth position of each object was used for MOT to examine the performance of MOT techniques regarding the position of real objects. It was noticed that our proposed approach outperformed the other MOT techniques. This was actualized by comparing all the positive and negative indexes for each prediction, as seen below.

4.2.1. Faster Region-Convolutional Network (Faster RCNN)

Table 1 shows the detection input of the faster RCNN evaluation with respect to all the performance metrics. Figure 4 represents the of performance comparison measures of IDF1, IDP, IDR and multiple object tracking precision (MOTP). The proposed method contained 55% of IDF1, 67% of IDP, 43% of IDR, 39% of multiple object tracking accuracy (MOTA), and 75% of MOTP. These measures were compared with the existing methods of SMOT, IOUT, GOG, CEM, SLSTM, SORT, RLSTM, RMOT, and the previously existing faster

RCNN. This statistical analysis proved that the proposed method had higher efficiencies than the existing methods.

Table 1. The detection input of the faster RCNN. MOT: multiple object tracking; IDP: identification precision; IDR: identification recall. MOTA: multiple object tracking accuracy. MOTP: multiple object tracking precision. FP: false positive. FN: false negative. IDS: identification switches, and the total number of times a trajectory is fragmented (FM).

MOT Methods	IDF1	IDP	IDR	MOTA	MOTP	FP	FN	IDS	FM
CEM	10.2	19.4	7	7.3	69.6	72,378	290,962	2488	4248
GOG	18	23.3	14.6	34.4	72.2	41,126	168,194	14,301	12,516
IOUT	23.7	30.3	19.5	36.6	72.1	42,245	163,881	9938	10,463
BMOT	33.3	27.8	41.4	39.8	72.3	319,008	151,485	5973	5897
RLSTM	31.3	38.6	26.3	25.6	69.1	71,955	180,461	1333	13,088
SMOT	45	55.7	37.8	33.9	72.2	57,112	166,528	1752	9577
SORT	43.7	58.9	34.8	39	74.3	33,037	172,628	2350	5787
SLSTM	37.2	46.8	30.8	37.9	72	44,783	161,009	6048	12,051
IPGAT	49.4	63.2	40.6	39	72.2	42,135	163,837	2091	10,057
PROPOSED	55	67	45	40.3	74	30,065	150,837	1091	3057

Figure 5 shows the proposed method's results in regards to the performance measures of FP, FN, IDS and FM. The proposed method contained 3000 FP, 13,000 FN, 2 IDR, and 5 FM values. These measures were compared with the existing methods of BMOT, IOUT, GOG, CEM, SLSTM, SORT, RLSTM, and RMOT in the faster RCNN. This statistical analysis proved that the proposed method of the minimum false rate had higher efficiencies compared to the existing methods.

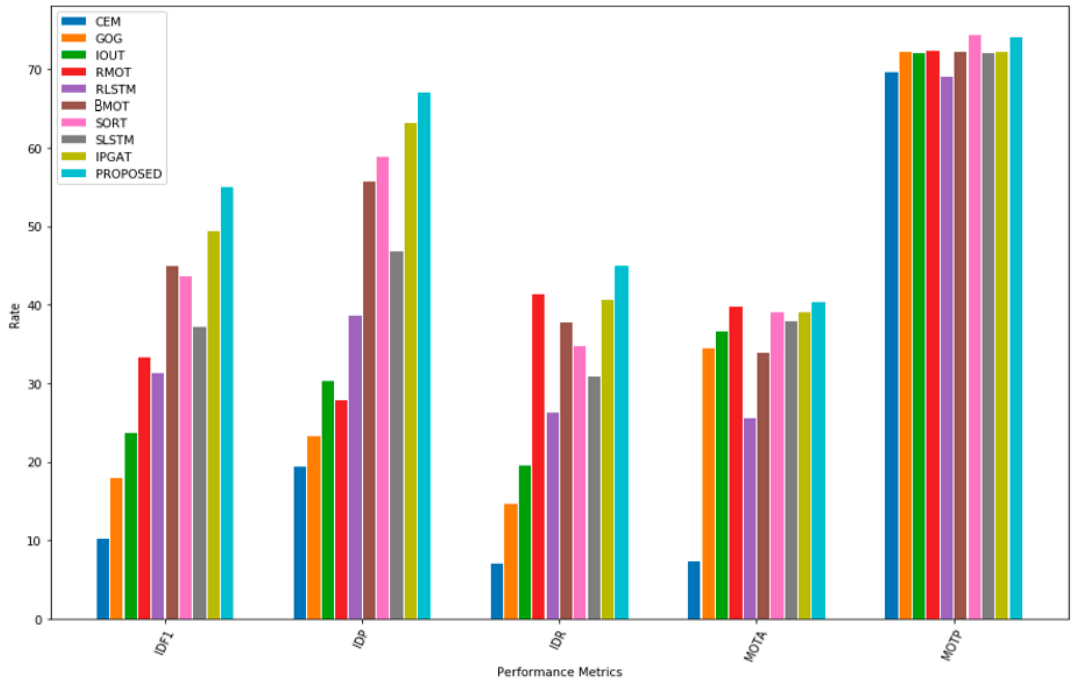


Figure 4. Comparative analysis of the proposed method with existing methods. CEM: continuous energy minimization, GOG: global optimal greedy, IOUT: intersection-over-union tracker, RMOT: relative motion online tracking, RLSTM: relative long short term memory, Bayesian multi-object tracking (BMOT), SORT: simple online and real-time tracking, SLSTM: social long short term memory.

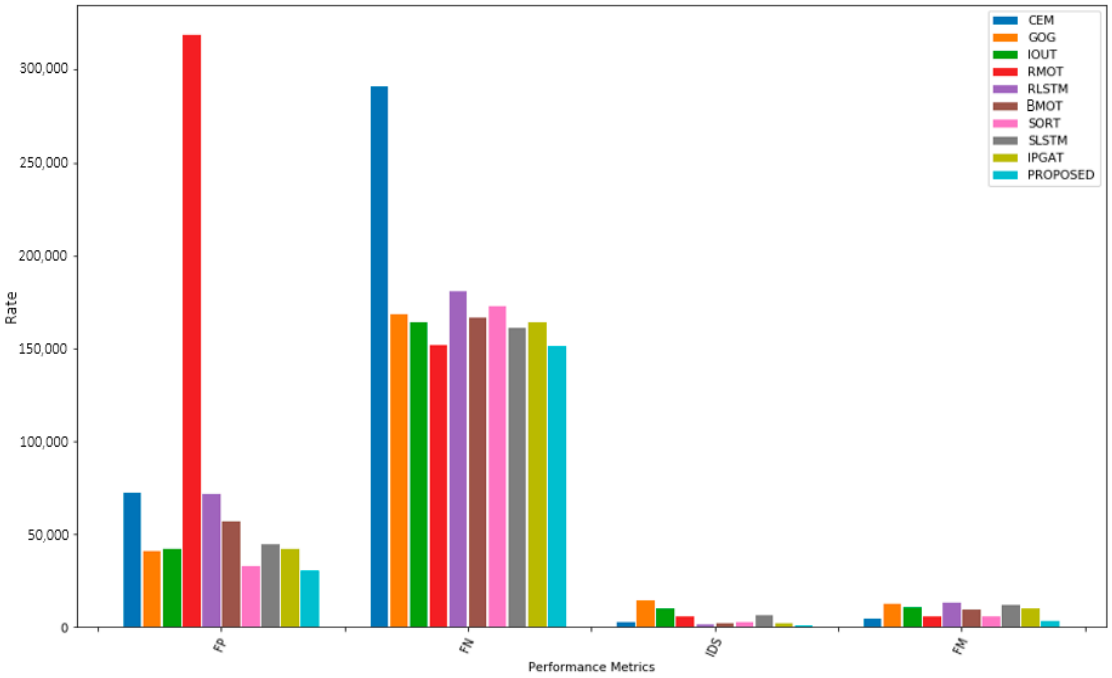


Figure 5. Comparative analysis of proposed method with existing methods based on the minimum false rate. CEM: continuous energy minimization, GOG: global optimal greedy, IOUT: intersection-over-union tracker, RMOT: relative motion online tracking, RLSTM: relative long short term memory, Bayesian multi-object tracking (BMOT), SORT: simple online and real-time tracking, SLSTM: social long short term memory.

4.2.2. The Faster Region-Based Fully Convolutional Network (Faster R-FCN)

Table 2 shows the detection input of the faster R-FCN with respect to all the performance metrics. Figure 6 shows the performance comparison measures of IDF1, IDP, IDR, and MOTP. The proposed method contained 50% IDF1, 64% IDP, 44% IDR, 40% MOTA, and 80% MOTP values. These measures were compared with the existing methods of SMOT, IOUT, GOG, CEM, SLSTM, SORT, RLSTM, and RMOT in the existing faster R-FCN. This statistical analysis proved that the proposed method had a higher accuracy and precision rate compared to the existing methods.

Table 2. R-FCN (region-based fully convolutional network) evaluation with respect to performance metrics. MOT: multiple object tracking; IDP: identification precision; IDR: identification recall. MOTA: multiple object tracking accuracy. MOTP: multiple object tracking precision. FP: false positive. FN: false negative. IDS: identification switches, and the total number of times a trajectory is fragmented (FM).

MOT Methods	IDF1	IDP	IDR	MOTA	MOTP	FP	FN	IDS	FM
CEM	10.3	18.4	7.2	9.6	70.4	81,617	289,683	2201	3789
GOG	30.9	38.8	25.6	28.5	77.1	60,511	176,256	6935	6823
IOUT	44	47.5	40.9	26.9	75.9	98,789	145,617	4903	6129
RMOT	34.4	28.7	43	43.8	76.3	328,677	158,760	2949	2286
RLSTM	29.3	34.8	25.2	14.7	70.6	97,670	191,720	1395	9953
SMOT	44	53.5	37.3	24.5	77.2	76,544	179,609	1370	5142
SORT	42.6	58.7	33.5	30.2	78.5	44,612	190,999	2248	4378
SLSTM	35.6	43.6	30.1	27.5	76.9	66,980	172,942	7355	9791
IPGAT	47.5	60.1	39.2	30.2	77.1	58,875	177,304	1799	6705
PROPOSED	49.5	64	44.2	40.3	79.6	42,065	135,617	1191	2057

Figure 7 shows the proposed method's results regarding the performance measures of FP, FN, IDS, and FM. The proposed method contained 42,065 FP, 13, 5617 FN, 1191 IDR, and 2057 FM values. These measures were compared with the existing methods of SMOT, IOUT, GOG, CEM, SLSTM, SORT, RLSTM, and RMOT in the existing faster SSD. This

statistical analysis proved that the proposed method had a higher accuracy and precision rate than the existing methods.

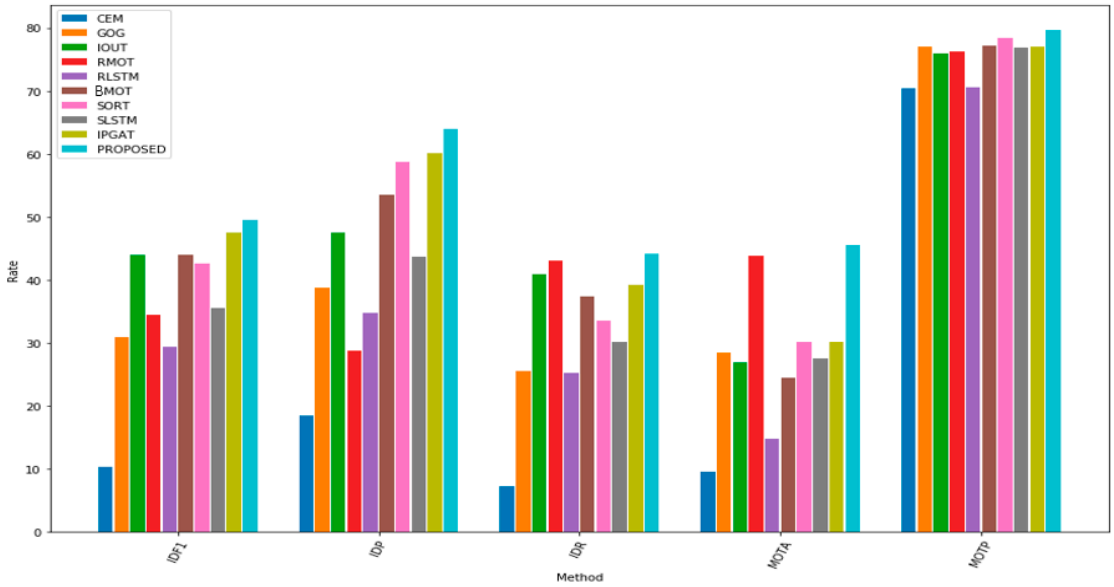


Figure 6. Comparative analysis of proposed method with existing methods. CEM: continuous energy minimization, GOG: global optimal greedy, IOUT: intersection-over-union tracker, RMOT: relative motion online tracking, RLSTM: relative long short term memory, Bayesian multi-object tracking (BMOT), SORT: simple online and real-time tracking, SLSTM: social long short-term memory.

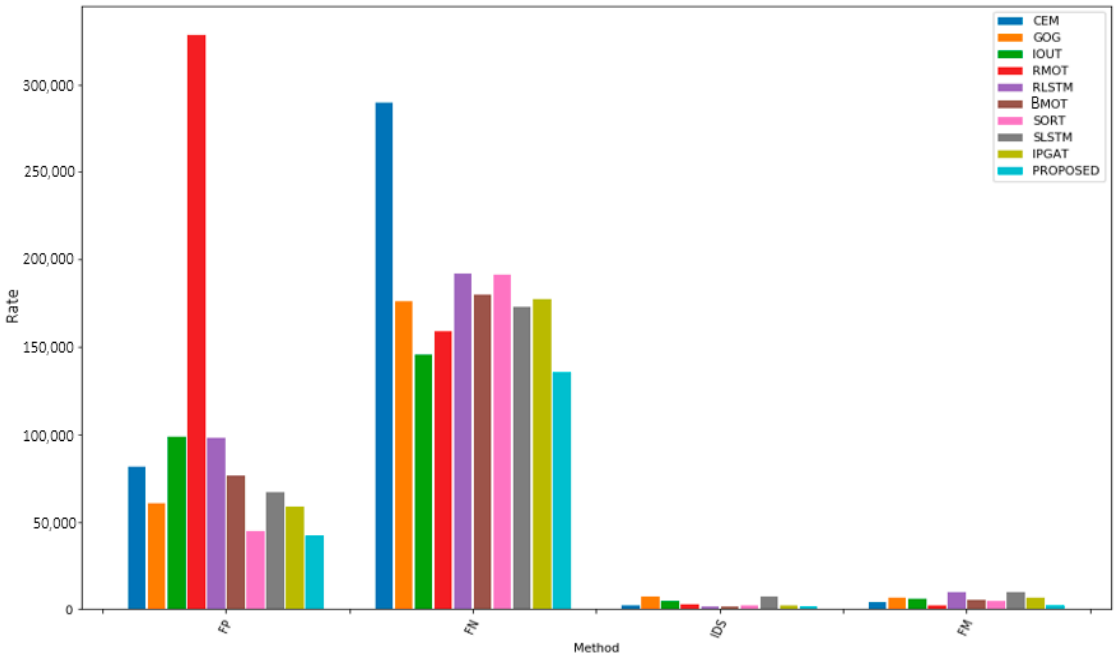


Figure 7. Comparative analysis of proposed method with existing methods. CEM: continuous energy minimization, GOG: global optimal greedy, IOUT: intersection-over-union tracker, RMOT: relative motion online tracking, RLSTM: relative long short term memory, Bayesian multi-object tracking (BMOT), SORT: simple online and real-time tracking, SLSTM: social long short term memory.

4.2.3. Single Shot Multi Box Detector (SSD)

Table 3 shows the detection input of SSD evaluation with respect to all the performance metrics.

Table 3. SSD (single-shot detector) evaluation with respect to performance metrics. MOT: multiple object tracking; IDP: identification precision; IDR: identification recall. MOTA: multiple object tracking accuracy. MOTP: multiple object tracking precision. FP: false positive. FN: false negative. IDS: identification switches, and the total number of times a trajectory is fragmented (FM).

MOT Methods	IDF1	IDP	IDR	MOTA	MOTP	FP	FN	IDS	FM
CEM	10.1	21.1	6.6	6.8	70.4	64,373	298,090	1530	2835
GOG	29.2	33.6	25.9	33.6	76.4	70,080	148,369	7964	10,023
IOUT	29.4	34.5	25.6	33.5	76.6	65,549	154,042	6993	8793
RMOT	26.1	18.9	41.8	10	75.1	544,591	131,382	9252	7211
RLSTM	26.5	28.9	24.4	7.6	69	129,660	182,828	347	12,654
SMOT	41.9	45.9	38.6	27.2	76.5	95,737	149,777	2738	9605
SORT	37.1	45.8	31.1	33.2	76.7	57,440	166,493	3918	7898
SLSTM	39.4	44.2	35.5	33	75.8	77,706	144,617	6019	13,332
IPGAT	43.3	49.6	38.5	34.1	76	71,519	148,248	4739	11,128
PROPOSED	44.5	51	42.2	35.5	78.6	54,065	129,617	1267	1890

Figure 8 shows the proposed method's results regarding the performance measures of IDF1, IDP, IDR, and MOTP. The proposed method contained 45% IDF1, 51% IDP, 42% IDR, 36% MOTA, and 79% MOTP values.

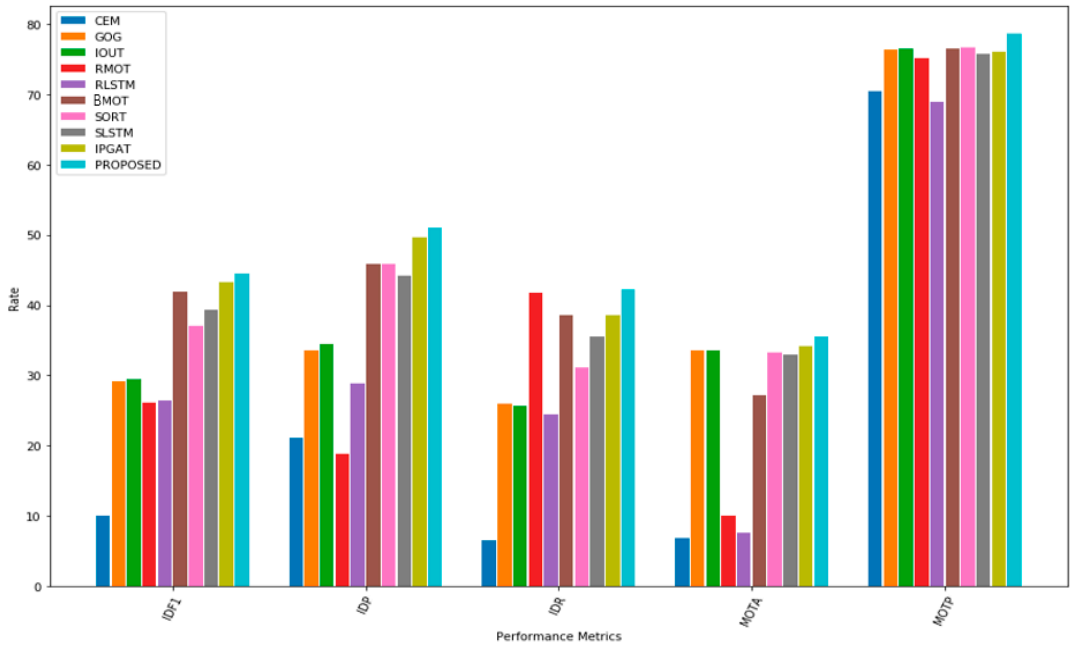


Figure 8. Comparative analysis of proposed method with existing methods. CEM: continuous energy minimization, GOG: global optimal greedy, IOUT: intersection-over-union tracker, RMOT: relative motion online tracking, RLSTM: relative long short term memory, Bayesian multi-object tracking (BMOT), SORT: simple online and real-time tracking, SLSTM: social long short term memory.

These measures were compared with the existing methods of SMOT, IOUT, GOG, CEM, SLSTM, SORT, RLSTM, and RMOT in the existing faster SSD. This statistical analysis proved that the proposed method had a higher accuracy and precision rate than the existing methods.

Figure 9 shows the proposed method's results regarding the performance measures of FP, FN, IDS, and FM. The proposed method contained 54,065 FP, 12, 9617 FN, 1267 IDR, and 1890 FM values. These measures were compared with the existing methods of SMOT, IOUT, GOG, CEM, SLSTM, SORT, RLSTM, and RMOT in the existing SSD. This statistical analysis proved that the proposed method of the minimum false rate had higher efficiencies than the existing methods.

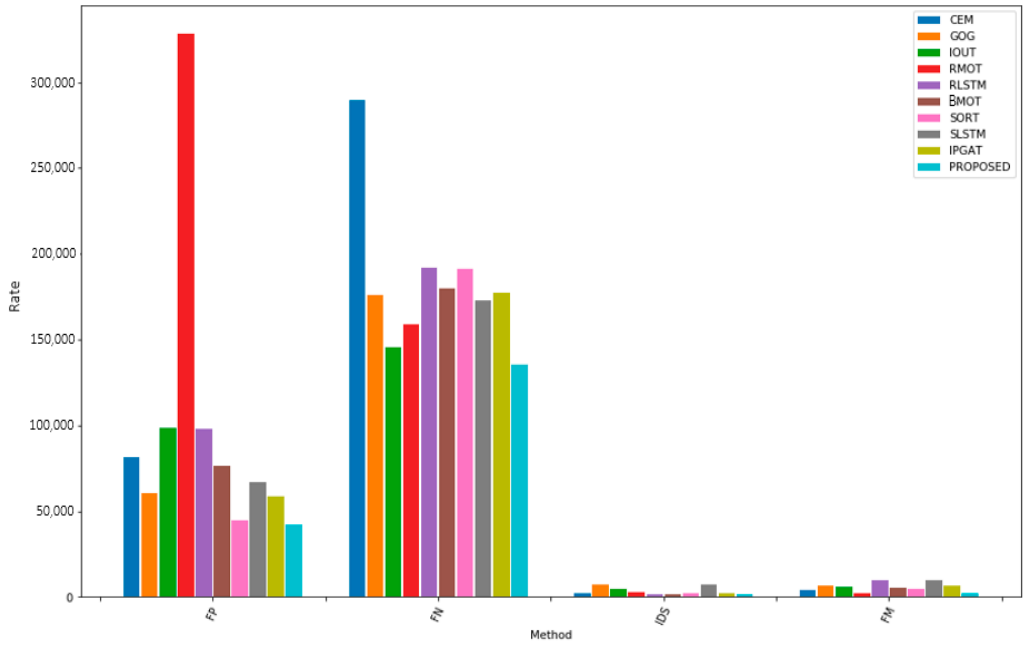


Figure 9. Comparative analysis of proposed method with existing methods. CEM: continuous energy minimization, GOG: global optimal greedy, IOUT: intersection-over-union tracker, RMOT: relative motion online tracking, RLSTM: relative long short term memory, Bayesian multi-object tracking (BMOT), SORT: simple online and real-time tracking, SLSTM: social long short term memory.

4.2.4. Reverse Connection with Objectiveness Prior Network (RON)

Table 4 shows the detection input of RON evaluation with respect to all the performance metrics.

Table 4. RON evaluation with respect to performance metrics. MOT: multiple object tracking; IDP: identification precision; IDR: identification recall. MOTA: multiple object tracking accuracy. MOTP: multiple object tracking precision. FP: false positive. FN: false negative. IDS: identification switches, and the total number of times a trajectory is fragmented (FM).

MOT Methods	IDF1	IDP	IDR	MOTA	MOTP	FP	FN	IDS	FM
CEM	10.1	18.8	6.9	9.7	68.8	78,265	293,576	2086	3526
GOG	51	60.2	44.3	35.7	72	62,929	153,336	3104	5130
IOUT	50.1	59.1	43.4	35.6	72	63,086	153,348	2991	5103
RMOT	28.8	22.5	40	34	70.9	418,222	153,480	7902	7007
RLSTM	36.5	42.4	32.1	24.1	68.7	87,589	169,866	1156	12,657
SMOT	52.6	60.8	46.3	32.8	72	73,226	154,696	1157	4643
SORT	54.6	66.9	46.1	37.2	72.2	53,435	159,347	1369	3661
SLSTM	48.1	56.4	41.9	35	71.9	65,093	152,481	4013	6059
IPGAT	54.1	64.3	46.8	35.9	72	62,038	154,871	1679	5062
PROPOSED	57.5	67	47.3	39.2	73.9	52,065	150,917	1476	2539

Figure 10 shows the proposed method's results regarding the performance measures of IDF1, IDP, IDR, and MOTP. The proposed method contained 58% IDF1, 67% IDP, 47% IDR, and 39% MOTA values. These measures were then compared with the existing methods of GOG, CEM, SLSTM, SORT, RLSTM, and RMOT in the previously existing faster RON.

Figure 11 shows the proposed method's results regarding the performance measures of FP, FN, IDS, and FM. The proposed method contained 52,065 FP, 150,917 FN, 1476 IDR, and 2539 FM values. These measures were compared with the existing methods of SMOT, IOUT, GOG, CEM, SLSTM, SORT, RLSTM, and RMOT in RON. This statistical analysis proved that the proposed method of the minimum false rate had higher efficiencies than the existing methods. We also evaluated our tracker with one-pass evaluation (OPE), and

our tracker was found to attain the best expected average overlap (EAO) while maintaining beyond real-time speed.

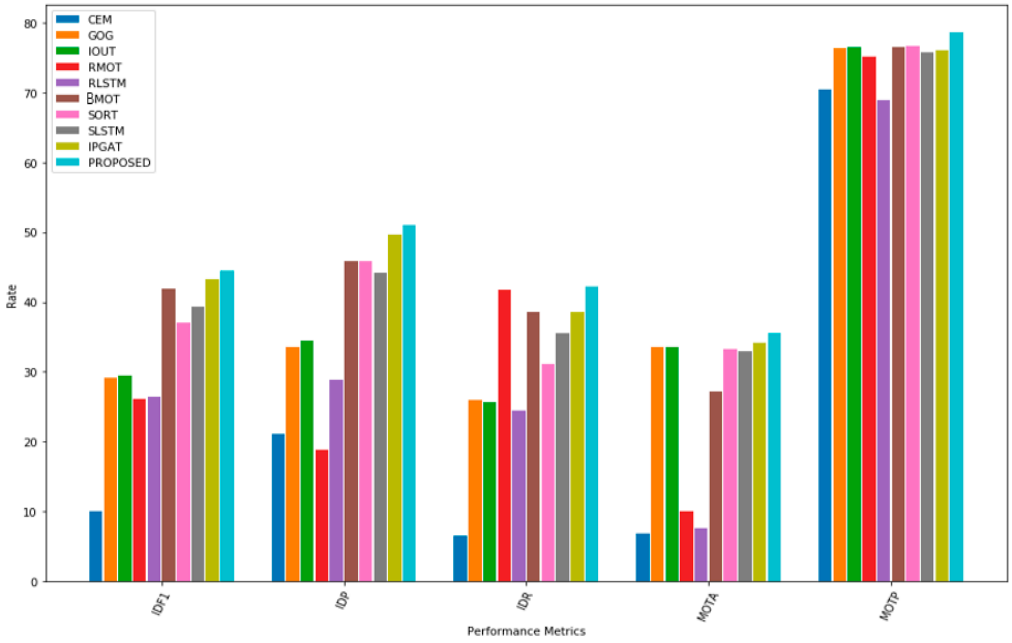


Figure 10. Comparative analysis of proposed method with existing methods. CEM: continuous energy minimization, GOG: global optimal greedy, IOUT: intersection-over-union tracker, RMOT: relative motion online tracking, RLSTM: relative long short term memory, Bayesian multi-object tracking (BMOT), SORT: simple online and real-time tracking, SLSTM: social long short term memory.

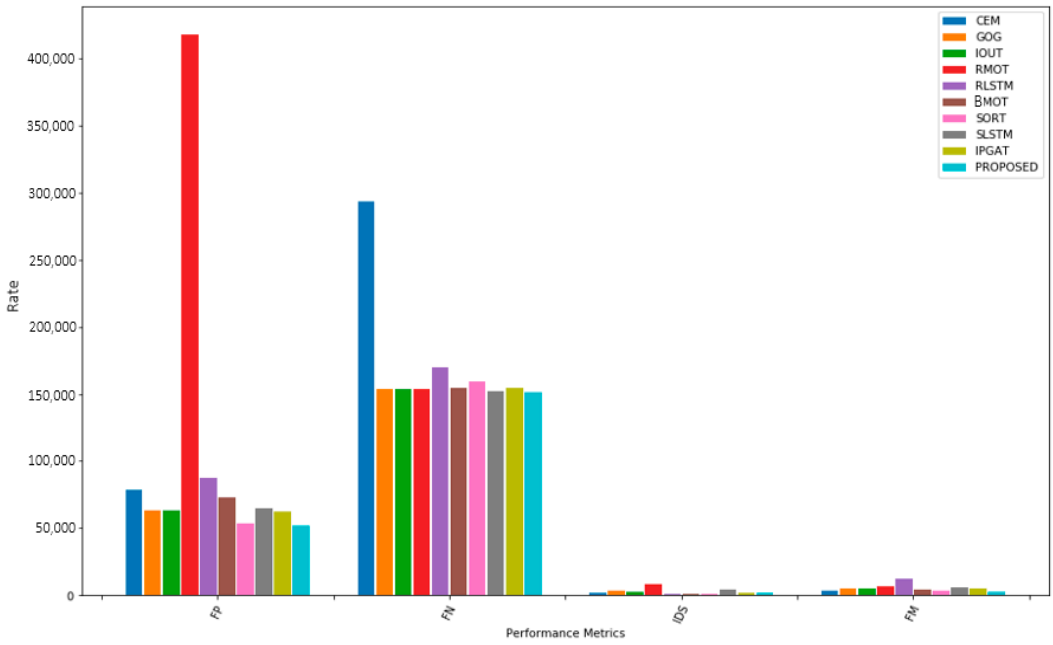


Figure 11. Comparative analysis of proposed method with existing methods. CEM: continuous energy minimization, GOG: global optimal greedy, IOUT: intersection-over-union tracker, RMOT: relative motion online tracking, RLSTM: relative long short term memory, Bayesian multi-object tracking (BMOT), SORT: simple online and real-time tracking, SLSTM: social long short term memory.

5. Conclusions

For MOT, motion is generally utilized as a cue for linking objects and predictions. However, the conventional MOT technique addresses a performance drop in UAV videos due to UAVs' autonomous motion. Therefore, we proposed a new quadruplet-based RCNN framework to find every objects' motion and view modifications of UAVs. Experiments were conducted on two public datasets (Stanford Drone and UAVDT), and the results show the efficiency of the proposed approach. Our future research work will include a combination of other modules such as a triplet network for abnormal event detection and tracking.

Author Contributions: Conceptualization, H.U.D.; methodology, H.U.D.; software, H.U.D.; validation, H.U.D. and Y.Z.; formal analysis, H.U.D. and Y.Z.; investigation, H.U.D. and Y.Z.; re-sources, H.U.D. and Y.Z.; data curation, H.U.D.; writing—original draft preparation, H.U.D.; writing—review and editing, H.U.D. and Y.Z.; visualization, H.U.D. and Y.Z.; supervision, Y.Z.; project administration, Y.Z.; funding acquisition, Y.Z. Both authors have read and agreed to the published version of the manuscript.

Funding: This work was supported under the National Key Research and Development Program of China (2018YFB1305505), National Natural Science Foundation of China (NSFC) (61973296), Shenzhen Basic Research Program Refs. JCYJ20180507182619669 and Science and Technology Planning Project of Guangdong Province, Ref. 2017B010117009.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhong, B.; Bai, B.; Li, J.; Zhang, Y.; Fu, Y. Hierarchical Tracking by Reinforcement Learning-Based Searching and Coarse-to-Fine Verifying. *IEEE Trans. Image Process.* **2019**, *28*, 2331–2341. [CrossRef]
2. Wang, X. Intelligent multi-camera video surveillance: A review. *Pattern Recognit. Lett.* **2013**, *34*, 3–19. [CrossRef]
3. Leal-Taixe, L.; Canton-Ferrer, C.; Schindler, K. Learning by Tracking: Siamese CNN for Robust Target Association. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 418–425.
4. Milan, A.; Leal-Taixé, L.; Reid, I.; Roth, S.; Schindler, K. MOT16: A benchmark for multi-object tracking. *Comput. Vis. Pattern Recognit.* **2016**, arXiv:1603.00831.
5. Milan, A.; Rezatofighi, S.H.; Dick, A.; Reid, I.; Schindler, K. Online multi-target tracking using recurrent neural networks. *AAAI Tech. Track Vis.* **2016**, arXiv:1604.03635.
6. Zhou, Q.; Zhong, B.; Zhang, Y.; Li, J.; Fu, Y. Deep Alignment Network Based Multi-Person Tracking with Occlusion and Motion Reasoning. *IEEE Trans. Multimed.* **2019**, *21*, 1183–1194. [CrossRef]
7. Yu, H.; Qin, L.; Huang, Q.; Yao, H. Online multiple object tracking via exchanging object context. *Neurocomputing* **2018**, *292*, 28–37. [CrossRef]
8. Luo, W.; Xing, J.; Milan, A.; Zhang, X.; Liu, W.; Kim, T.-K. Multiple object tracking: A literature review. *Artif. Intell.* **2021**, *293*, 103448. [CrossRef]
9. Comaniciu, D.; Ramesh, V.; Meer, P. Kernel-based object tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 564–577. [CrossRef]
10. Hu, W.; Tan, T.; Wang, L.; Maybank, S. A Survey on Visual Surveillance of Object Motion and Behaviors. *IEEE Trans. Syst. ManCybern. Part. C Appl. Rev.* **2004**, *34*, 334–352. [CrossRef]
11. Choi, W.; Savarese, S. A Unified Framework for Multi-target Tracking and Collective Activity Recognition. In Proceedings of the Lecture Notes in Computer Science, Florence, Italy, 7–13 October 2012; pp. 215–230.
12. Hong, I.; Kuby, M.; Murray, A. A deviation ow refueling location model for continuous space: A commercial drone de-livery system for urban areas. In Proceedings of the 13th International Conference on Advances in Geocomputation, Geocomputation, Dallas, TX, USA, 20–23 May 2017; pp. 125–132.
13. Paul, M.; Haque, S.M.E.; Chakraborty, S. Human detection in surveillance videos and its applications-a review. *Eurasip J. Adv. Signal. Process.* **2013**, *2013*, 176. [CrossRef]
14. Lee, K.S.; Ovinis, M.; Nagarajan, T. Autonomous patrol and surveillance system using unmanned aerial vehicles. In Proceedings of the International Conference on Environment and Electrical Engineering (EEEIC), Rome, Italy, 10–13 June 2015; pp. 1291–1297.
15. Bochinski, E.; Eiselein., V.; Sikora., T. High-speed tracking-by-detection without using image information. In Proceedings of the 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (ICAVSS), Lecce, Italy, 29 August–1 September 2017; pp. 1–6. [CrossRef]
16. Wojke, N.; Bewley, A.; Paulus, D. Simple online and realtime tracking with a deep association metric. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2017; pp. 3645–3649.

17. Chen, L.; Ai, H.; Zhuang, Z.; Shang, C. Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In Proceedings of the 2018 IEEE International Conference on Multimedia and Expo (ICME), San Diego, CA, USA, 23–27 July 2018; Volume 5, p. 8.
18. Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6154–6162.
19. Sun, D.; Yang, X.; Liu, M.Y.; Kautz, J. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8934–8943.
20. Zhu, Z.; Wang, Q.; Li, B.Q.; Wu, W.; Yan, J.; Hu, W. Distractor-Aware Siamese Networks for Visual Object Tracking. In Proceedings of the ECCV, Munich, Germany, 8–14 September 2018; pp. 101–117.
21. Bewley, A.; Zongyuan, G.; Ott, L.; Ramos, F.; Upcroft, B. Simple online and realtime tracking. In Proceedings of the IIEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 3464–3468.
22. Son, J.; Baek, M.; Cho, M.; Han, B. Multi-object Tracking with Quadruplet Convolutional Neural Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 3786–3795.
23. Dong, X.; Shen, J.; Wu, D.; Guo, K.; Jin, X.; Porikli, F. Quadruplet Network with One-Shot Learning for Fast Visual Object Tracking. *IEEE Trans. Image Process.* **2019**, *28*, 3516–3527. [CrossRef]
24. Sun, S.; Akhtar, N.; Song, H.; Mian, A.S.; Shah, M. Deep Affinity Network for Multiple Object Tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *1*. [CrossRef]
25. Xu, Y.; Wang, J. A unified neural network for object detection, multiple object tracking and vehicle re-identification. *Comput. Vis. Pattern Recognit.* **2019**, arXiv:1907.03465.
26. Hou, J.; Zeng, H.; Zhu, J.; Hou, J.; Chen, J.; Ma, K.-K. Deep quadruplet appearance learning for vehicle re-identification. *IEEE Trans. Vehi. Tech.* **2019**, *1*, 1–9. [CrossRef]
27. Hung, G.L.; Sahimi, M.S.B.; Samma, H.; Almohamad, T.A.; Lahasan, B. Faster R-CNN Deep Learning Model for Pe-destrian Detection from Drone Images. *SN Comp. Sci.* **2020**, *1*, 1–9.
28. Yu, H.; Li, G.; Zhang, W.; Huang, Q.; Du, D.; Tian, Q.; Sebe, N. The Unmanned Aerial Vehicle Benchmark: Object Detection, Tracking and Baseline. *Int. J. Comput. Vis.* **2019**, *128*, 1141–1159. [CrossRef]
29. Li, L.-Q.; Wang, X.-L.; Liu, Z.-X.; Xie, W.-X. A Novel Intuitionistic Fuzzy Clustering Algorithm Based on Feature Selection for Multiple Object Tracking. *Int. J. Fuzzy Syst.* **2019**, *21*, 1613–1628. [CrossRef]
30. Meng, F.; Wang, X.; Wang, D.; Shao, F.; Fu, L. Spatial–Semantic and Temporal Attention Mechanism-Based Online Multi-Object Tracking. *Sensors* **2020**, *20*, 1653. [CrossRef]
31. Lu, X.; Ma, C.; Ni, B.; Yang, X. Adaptive Region Proposal with Channel Regularization for Robust Object Tracking. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *1*. [CrossRef]
32. Wan, S.; Chen, Z.; Zhang, T.; Zhang, B.; Wong, K.-K. Bootstrapping face detection with hard negative examples. *Comput. Vis. Pattern Recognit.* **2016**, arXiv:1608.02236.
33. Xiao, Q.; Luo, H.; Zhang, C. Margin sample mining loss: A deep learning based method for person re-identification. *Comput. Vis. Pattern Recognit.* **2017**, arXiv:1710.00478.
34. Dicle, C.; Camps, O.I.; Sznaier, M. The Way They Move: Tracking Multiple Targets with Similar Appearance. In Proceedings of the 2013 IEEE International Conference on Computer Vision; Institute of Electrical and Electronics Engineers (IEEE), Sydney, Australia, 1–8 December 2013; pp. 2304–2311. [CrossRef]
35. Pirsiavash, P.; Ramanan, D.; Fowlkes, C.C. Globally-optimal greedy algorithms for tracking a variable number of objects. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, CO, USA, 20–25 June 2011; pp. 1201–1208.
36. Milan, A.; Roth, S.; Schindler, K. Continuous Energy Minimization for Multitarget Tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 58–72. [CrossRef] [PubMed]
37. Alahi, A.; Goel, K.; Ramanathan, V.; Robicquet, A.; Fei-Fei, L.; Savarese, S. Social LSTM: Human Trajectory Prediction in Crowded Spaces. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); Institute of Electrical and Electronics Engineers (IEEE), Las Vegas, NV, USA, 27–30 June 2016; pp. 961–971. [CrossRef]
38. Yoon, J.H.; Yang, M.-H.; Lim, J.; Yoon, K.-J. Bayesian Multi-object Tracking Using Motion Context from Multiple Objects. In Proceedings of the 2015 IEEE Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 5–9 January 2015; pp. 33–40. [CrossRef]
39. Robicquet, A.; Sadeghian, A.; Alahi, A.; Savarese, S. Learning Social Etiquette: Human Trajectory Understanding IN Crowded Scenes. *Lect. Notes Comput. Sci.* **2016**, 549–565. [CrossRef]
40. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Patt. Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
41. Kong, T.; Sun, F.; Yao, A.; Liu, H.; Lu, M.; Chen, Y. RON: Reverse Connection with Objectness Prior Networks for Object Detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Honolulu, HI, USA, 21–26 July 2017; pp. 5244–5252. [CrossRef]
42. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y. Ssd: Single shot multibox detector. In Proceedings of the 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 21–37.