IET Computer Vision

Special Issue: Computer Vision for Smart Cameras and Camera Networks

IET Journals
The Institution of Engineering and Technology

Harnessing feedback region proposals for multi-object tracking

ISSN 1751-9632
Received on 30th November 2019
Revised 12th June 2020
Accepted on 7th July 2020
E-First on 14th October 2020
doi: 10.1049/iet-cvi.2019.0943
www.ietdl.org

Aswathy Prasanna Kumar[1] ✉, *Deepak Mishra*[1]

[1]*Department of Avionics, Indian Institute of Space Science and Technology, Trivandrum, Kerala, India*

✉ *E-mail: aswathyece2011@gmail.com*

Abstract: In the tracking-by-detection approach of online multiple object tracking (MOT), a major challenge is how to associate object detections on the new video frame with previously tracked objects. Two important aspects that directly influence the performance of MOT are quality of detection and accuracy in data association. The authors propose an efficient and unified MOT framework for improved object detection, followed by enhanced object tracking. The object detection and tracking are considered as two independent functions in the tracking-by-detection paradigm. In this study, object detection accuracy has been increased by employing a faster region-based convolutional neural network (Faster R-CNN) modified with the feedback region proposals from the tracker. Target association is performed by the correlation filter-based Siamese CNN model, which finds the similarity score between the input image patches. The Siamese CNN is trained using a supervised hard sample mining strategy. An optical flow-based motion model is employed to predict the next probable location of the targets from the tracker and these region proposals are fed back to the classifier module of Faster R-CNN. The authors' extensive analysis of publicly available MOT benchmark datasets and comparison with the state-of-the-art tracking methods demonstrate competitive tracking performance of the proposed MOT framework.

1 Introduction

Multiple object tracking (MOT) is the process of localising multiple moving objects over time. The problem of tracking multiple objects in a video sequence poses several challenging tasks, including estimation of the time-varying number of objects, motion prediction of all objects, object re-identification and dealing with long and short term occlusions. A common approach for solving the multi-object visual tracking problem is tracking-by-detection. In tracking-by-detection paradigm of MOT: first, an object detector is applied to each frame of the video to locate the objects of interest, then using data association algorithms, a unique identity is assigned to every detected object. These identities are linked across a sequence of frames to form object trajectories.

Online and batch methods are two commonly adopted methods for trajectory extraction. Online methods [1, 2] use the current and previous frames detections to object state estimation at each time epoch. The batch techniques [3–5] require the observations from a batch of frames in advance to estimate the final object state. Therefore, batch methods are difficult to use reliably in real-time applications. The proposed MOT-by-detection framework works in online fashion.

In MOT, generally, the object detector and the tracker are considered as two independent modules where the detection responses need to be reliably linked to form target trajectories. However, the performance of the tracker heavily relies on the quality of detection results from the object detector. Recent advancements in deep-learning-based object detection systems [6–9] have improved the MOT performance significantly. In the proposed MOT framework, the highly efficient Faster R-CNN (faster region-based convolutional neural network) [8] is used as the person detector module. However, the deployed detector module does not consider temporal information while detecting the object in the respective frames. Hence, in order to study the effect of temporal feedback on MOT, in the proposed method, feedback region proposals from the past object state estimation are given to the object detector. This introduction of feedback helps to reduce the missed detections in the video sequences and thus improves the overall MOT accuracy.

Once the object detections are obtained from the detector, what matters the most is how to associate the current detections with the existing tracks. If there is a missing or inaccurate detection, the target is prone to be lost. To alleviate such issues, the proposed MOT approach integrates the merits of single object tracking and data association methods in a unified framework. A single-object tracker uses the detection in the first frame and updates the appearance model online to find the target in the subsequent frames. The data association method computes the similarity between the detections in the frame and tracklets from the previous frames. Recently, the application of the Siamese network [10–13] is found to be very useful for reliable and robust data association in MOT. In Siamese based architectures, the input is a pair of image patches. The network learns a similarity metric between the patches and outputs a similarity score between them. In the proposed method, a correlation filter-based Siamese CNN (CFNet) [11] is used in both roles; as a single-object tracker and as a data association method. CFNet is trained for data association using a supervised hard sample mining strategy. The hard positive and hard negative training samples for learning are generated according to the influence factor derived from the online MOT results with the supervision of ground truth trajectories.

In this paper, we introduce an efficient MOT framework that incorporates an improved object detector and an efficient data association algorithm. Additionally, to handle the missed detections, feedback region proposals from the tracker are presented to the object detector. Likewise, the data association problem is tackled by introducing a re-identification Siamese CNN model (CFNet). The proposed method also benefits from the strengths of CFNet as an online single-object tracker, where the target appearance model is updated online to handle the detection failures. In addition to this, the Siamese network is trained on a supervised learning strategy, which helps the convolutional neural network (CNN) model to learn the context in the MOT dataset. To assess the effectiveness of the proposed MOT framework, a comprehensive analysis is performed on the publicly available MOT benchmark datasets, 2DMOT15 and MOT17. The comparative results against some of the recent state-of-the-art methods show that our method performs substantially better in terms of common metrics used in the MOT literature.

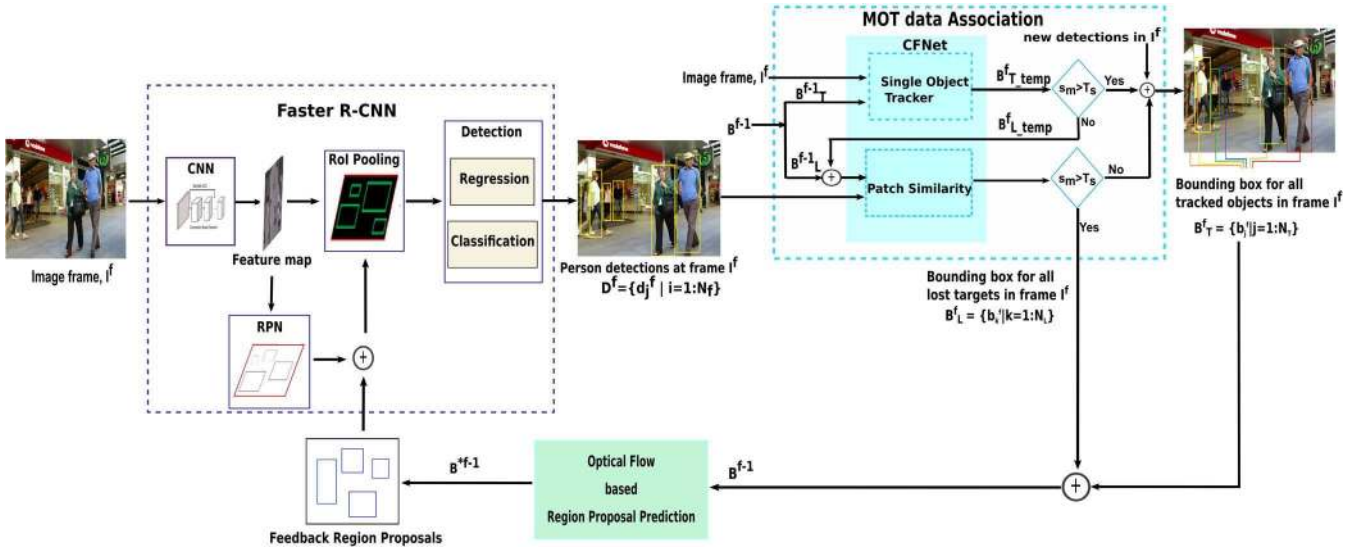

Fig. 1 *Faster R-CNN object detector with feedback region proposals from the MOT tracker. For each input frame I^f, Faster R-CNN detector outputs detection bounding boxes considering all the region proposals from the RPN module and feedback region proposals from the past trajectories. The data association module incorporates CFNet functions as a single object tracker that outputs the location of the tracked targets and as a patch similarity metric that gives a similarity score for the detection assignment of lost targets*

The remaining of the paper is organised as follows: In Section 2, we review the background on the MOT, object detection methods and the Siamese CNN models. Section 3 describes the proposed tracking framework in detail. The employed Faster R-CNN module with the feedback region proposals framework is briefly explained in Section 3.1 and the data association methodology that incorporates the CFNet is discussed in Section 3.2. The training procedure adopted for training CFNet for data association in the proposed MOT framework is discussed in Section 3.3. Section 3.4 summarises the proposed MOT algorithm. Experimental evaluations and results are detailed in Section 4. Finally, we conclude the paper with a brief discussion on possible future directions in Section 5.

2 Related works

MOT: Despite the recent advances in MOT, it remains a complex and difficult task in crowded environments with frequent occlusions, similar appearance, false detections, etc. Mainly the MOT methods can be classified into three categories; (i) the data association problem modelled as an optimisation problem or graphs [14, 15], (ii) solve data association problem using an end-to-end neural network [16, 17], (iii) use MOT paradigm other than tracking-by-detection [18]. The first two categories give a solution with a tracking-by-detection approach, where the detector and tracker exist as two independent functions. Most of the recent MOT trackers follow the tracking-by-detection paradigm [17–20]. In our framework, these two modules co-exist and the feedback from the tracker is given to the detector in every frame. Finally, the third category aims to search for novel and more simple MOT methods, but the trade-off between performance and speed still needs improvement.

Object detection: Most of the recent multiple object trackers follow the tracking-by-detection approach, which heavily depends on the quality of object detection. Initially, most of the object detectors relied on using handcrafted features. But the introduction of deep CNNs has demonstrated remarkable improvement in the performance of object detectors. In [21], the idea of selective search is employed for proposing probable object locations. The recent development in object detection is driven by the success of region-based CNNs (R-CNNs) [22]. Advances like SPPNet [6] and Fast R-CNN [7] have improved the detection performance with reduced running time. The Faster R-CNN detector [8] and further SDP detector [23] employ a fully CNN for region proposals and classification without any handcrafted features. Then Redmon *et al.* [9] proposed You Only Look Once (YOLO) detector that bypasses the need for a region proposal network.

Siamese CNNs: In MOT, data association can be addressed using similarity learning between image patches. Siamese CNN is widely used for similarity measurements and the CNN architecture used in the model gives a better image feature representation. Bertinetto *et al.* [10] proposed a fully convolutional Siamese network to measure the similarity score between image patches, which is employed for object tracking. CFNet [11] is an asymmetric architecture that incorporates a correlation filter into the Siamese network. Other variants to Siamese CNN include DSiam [24] that uses fast transfer motion to update the model, SINT [25] that makes use of optical flow methods and SA-Siam [26] that utilises the combination of original Siamese architecture. In [13], an end-to-end trainable Siamese region proposal network is introduced for object tracking that includes a Siamese network for feature extraction and a region proposal network with two branches, one for foreground–background classification and the other for proposal regression. In the proposed MOT framework, the Siamese network performs in two roles: as a single object tracker and as a similarity function.

3 Online MOT framework

We propose an online MOT framework that uses two popular CNN architectures that are tailored according to the MOT framework; in particular, we deploy Faster R-CNN [8] for person detection and correlation filter-based Siamese network (CFNet) [11] for data association. To enhance the accuracy of the detection and to reduce missed detections, here we introduce feedback region proposals from the tracker to the detector, using the optical flow based motion model of the target objects. In tracking-by-detection, the detections in the current frame are associated with the existing tracks using an efficient data association algorithm. In this study, we exploit the merits of both single object tracking and data association to maintain target identities in a unified MOT framework. The proposed MOT method benefits from the strengths of CFNet, which allows us to use it as an online single-object tracker and a similarity-based data association alternative. In the following sections, a detailed description of the proposed five online MOT algorithms is given.

3.1 Faster R-CNN detector with feedback region proposals

To perform an effective person detection, we employ Faster R-CNN architecture in our framework. The block diagram representation of the Faster R-CNN object detector with feedback region proposals from the MOT tracker is shown in Fig. 1. There are two main stages in the Faster R-CNN detector, a region

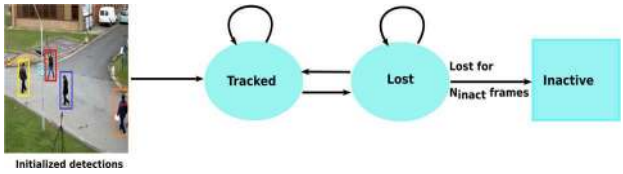

Fig. 2 *Target state transition in the proposed MOT framework. A target in each video frame can go through four different stages: initialised (detections from the object detector), tracked, lost and inactive (terminate the trajectory)*

proposal network (RPN) and a classifier-regressor module. The RPN generates a multitude of region proposals for each probable object of interest in the input image. The prerequisite step for RPN is feature extraction using a pre-trained convolution neural network. To generate good quality feature maps, we use ResNet-101 as our backbone CNN. RoI (Region of Interest) pooling method is used to extract the feature maps for the proposals in uniform scale and size. RoIs are then proceeded to the classification and bounding box regression modules. The classifier assigns a class score for each RoI and the regression module realigns the bounding box to fit with the object. The final set of detections is obtained after employing a non-maximum-suppression (NMS) step to the bounding boxes. In this study, we are interested only in the person category after detection.

Generally, in MOT, detection and data association are considered as two independent strategies. It is, therefore, the detector that does not require any temporal information of objects to perform its task. Here, we investigate the performance enhancement in an object detector when it is provided with temporal information about the past target trajectories. The probable locations of the existing targets (tracked and lost targets in the previous frame I^{f-1}) in the current frame I^{f} are predicted using its optical flow motion model and given as feedback to the detector as region proposals. Let B^{f-1} represents the set of target bounding boxes in the frame I^{f-1}, which includes the locations of tracked and lost targets.

$$\begin{aligned} &B^{f-1} = \left\{B_T^{f-1}, B_L^{f-1}\right\}, \\ &B_T^{f-1} = \{b_j^{f-1}\}_{j=1}^{N_T}; \quad B_L^{f-1} = \{b_k^{f-1}\}_{k=1}^{N_L}, \quad \text{where} \\ &\quad b_i = \{x, y, w, h\}\,. \end{aligned} \tag{1}$$

In (1), N_T and N_L are the numbers of tracked and lost targets, (x, y) is the centre coordinates of the target, and (w, h) are the width and height of the target, respectively. In order to estimate the new target location, we compute an optical flow from densely and uniformly sampled points inside the current target template to the new video frame. Specifically, given the current target position, $p = (x, y)$, we find its corresponding location $p^* = p + u = (x + u_x, y + u_y)$ in the new frame using the iterative Lucas–Kanade method with pyramids [27], where $u = (u_x, u_y)$ is the optical flow at a point p. We can predict the new bounding box for the target with centre p^* and size the same as the previous box (w, h), which is now treated as the region proposal for that target in the new frame. As shown in Fig. 1, the set of all the predicted feedback region proposals of the current targets $B^{*f-1} = \left\{B_T^{*f-1}, B_L^{*f-1}\right\}$, tracked and lost ones, are given back to the Faster R-CNN detector module. Along with the proposals from the RPN module of the detector, the feedback region proposals from the tracker are provided to the RoI pooling. The remaining flow of the Faster R-CNN is unchanged.

While evaluating the MOT framework, we observe that the two measures that impact the performance are identity switches and fragmentation related issues. The number of times a particular target changes its identity is measured by identity switches. Whereas, when the object is not detected in some frames, then fragmented trajectories are generated. These two issues occur mainly due to the missed detections in the video frames. The feedback proposals from the previous tracks in the proposed MOT algorithm helps to reduce the missed detections in each frame and thereby reduces the identity switches of the target and fragmented trajectories.

3.2 MOT – data association methodology

In this section, we extend our discussion on the proposed MOT framework, where we now incorporate the correlation filter-based Siamese network (CFNet) to tackle the data association problem. For each video frame, Faster R-CNN provides person detections. The data association algorithm identifies a correspondence between the new object detections and pre-existing tracks. Here, we integrate the merits of single-object tracking and CNN-based similarity metric for data association. The Siamese CNN performs better as both a single-object tracker and a similarity network.

In the proposed MOT framework, we adopt the state transitions of the target, as explained in [28] with some modifications. A target in the video can go through four different stages, such as initialised, tracked, lost and inactive. Fig. 2 illustrates these state transitions of the targets between the listed four stages. The object trajectory is initialised when an object appeared in the video frame for the first time. In the first frame, all the detections from the detector are considered as tracked targets and for each detection a new trajectory is initialised in the trajectory list. Now, the single-object tracker has to take a decision whether to keep each target in the tracked state or transfer it into the lost state. The state of the target is set as tracked until it is not occluded or is not out of the camera's field-of-view. Otherwise, the target is regarded as lost. This decision making is related to the tracking score and consistency of tracking results with the object detections. Once the object is transferred into the lost state, the data association algorithm tries to find out a match for the lost targets within the detections, that are not covered by any tracked target. If the similarity function could find a matched detection for the lost target, the state is updated as tracked and tracking process resumes for the same. If the target stays in the lost state for a long time (say N_{inact} as the number of frames for which the target is in the lost state), it is considered that the object entered into an inactive state and we terminate the trajectory corresponding to that object.

In the proposed scheme, the tracking problem is addressed by using a correlation filter-based Siamese CNN (CFNet) model that predicts whether two image patches belong to the same trajectory or not. The functions of the CFNet here are two-fold:

- If the target is in the tracked state, CFNet works as a single-object tracker that finds out the new target location.
- If the object is occluded, i.e. in the lost state, CFNet acts as a patch similarity function, that gives a similarity score between the lost target template and detections from the detector module that is not associated with tracked objects.

The pre-trained CFNet architecture with learned weights is adopted in the proposed MOT framework and is retrained with the MOT benchmark dataset on a supervised learning manner that uses a hard sample mining strategy.

3.2.1 CFNet as a single-object tracker: Visual object tracking algorithms based on a Siamese CNN architecture formulate tracking as a template matching problem [10, 11, 13, 25]. The network structure has two identical CNN branches that share the network weights. One branch extracts the feature maps of the target image patch and the other one of the search image patches, which contains the candidate objects. The search area is chosen with the centre, the same as the previous target location and size 2.5 times larger than the target image. A number of candidate image patches with the same size as the target are chosen within the search area. Then we obtain a similarity score map by the cross-correlation between the convolutional feature maps of target and candidate patches. In object tracking, the goal is to find the new target location and is obtained from the most similar candidate image patch.

Let x_T represents the target patch and x_c represents the candidate patch. These inputs are processed by CNN, ϕ_w, where w is the

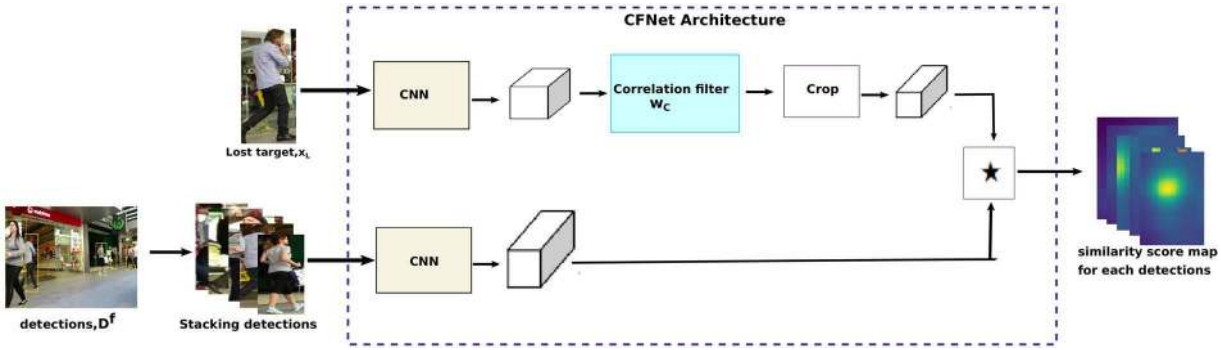

Fig. 3 *CFNet as the patch similarity metric in data association of the targets in the lost state. The CFNet cross-correlate the image patches; the lost target (x_L) and detections in the current frame (D^f) presented and generate the similarity score maps for each image pair (x_L, d_i^f). The detection with maximum similarity is considered to associate with the lost target*

learnable parameters. Then the feature maps $\phi_w(x_T)$ and $\phi_w(x_c)$ are cross-correlated ($\star$) as given by

$$\Psi_w(x_T, x_c) = \phi_w(x_T) \star \phi_w(x_c) \tag{2}$$

The correlation filter-based Siamese network (CFNet) [11] incorporates two additional layers within the baseline Siamese network [10], correlation filter and crop layers, which makes it shallower and faster without accuracy drop. The correlation filter layer inserted between the CNN with target patch and cross-correlation module estimates the discriminative features of the target patches. Then the modifications in (2) can be formulated as

$$\Psi_{w,\alpha,\beta}(x_T, x_c) = \alpha\Omega_c(\phi_w(x_T)) \star \phi_w(x_c) + \beta, \tag{3}$$

where $\Omega_c(.)$ represents the correlation filter layer that learns during training, by solving a ridge regression problem in the frequency domain by using Fourier transform. In this equation, α and β are scale and bias parameters, respectively. The maximum score in the similarity map corresponds to the new target location

$$\widehat{x_T} = \arg\max_{x_c^i} \Psi_{w,\alpha,\beta}(x_T, x_c^i) \tag{4}$$

3.2.2 CFNet for patch similarity: When the target is in the lost state, the data association algorithm has to decide whether to retain this target in the lost state, move into the tracked state or terminate the target trajectory (inactive state). In order to move it from the lost state to the tracked state, any of the detections from the person detector needs to be associated with this lost target. A Siamese network can also be used as a similarity function that checks the pairwise similarity between the lost target patch and the detections. Fig. 3 depicts how we use the CFNet architecture as the data association module by considering the pairwise patch similarity metric. To elaborate it further, let x_L represents the lost target image patch and $D^f = \{d_i^f\}_{i=1}^{N_f}$ are the set of detections given in the current frame, I^f. CFNet outputs N_f similarity score maps, each corresponding to the match score between each detection and the lost target. If the maximum similarity score s_m is above the threshold T_s, then the detection and lost target pair are considered for data association. The Hungarian algorithm is employed to assign the detections to the lost targets. If a detection is associated with the lost target then it is transferred to the tracked state and the detection corresponding to the maximum score, d_m^f is updated as the target image, $\widehat{x_L}$.

$$d_m^f = \arg\max_{d_i^f} \Psi_{w,\alpha,\beta}(x_L, d_i^f); \quad i = 1{:}N_f \tag{5}$$

$$s_m = \max\ (\Psi_{w,\alpha,\beta}(x_L, d_m^f)) \tag{6}$$

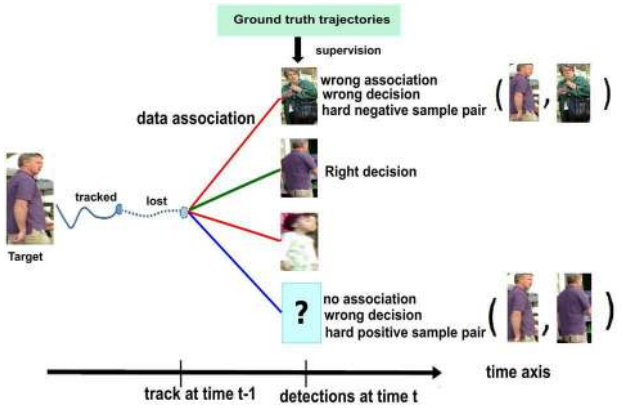

Fig. 4 *Supervised hard sample mining: hard positive and hard negative training samples are generated when the decision of the data association system went wrong. The image pairs (lost target, detection) with the wrong association are considered as hard negative samples. The image pairs (lost target, true detection) that missed the association are considered as hard positive samples*

$$\text{state} = \begin{cases} \text{tracked}, & \text{if } s_m \geq T_s \\ \text{lost}, & \text{otherwise} \end{cases} \tag{7}$$

$$\hat{x}_L = d_m^f; \quad \text{if state} = \text{tracked} \tag{8}$$

3.3 Training CFNet – supervised hard sample mining

In this study, the CFNet architecture, which is trained offline [11] is turned out to be a backbone Siamese CNN. Within the context of MOT, the weight parameters of the network are then retrained using a supervised hard sample mining strategy. In our framework, hard samples are the errors in the decision or false alarms from the trained CFNet system. Hard negative samples are the false positives from the CFNet in which similarity score wrongly indicates the presence of the target, while in reality, it is not present. Hard positive samples are the false negatives from the CFNet in which similarity score wrongly indicates the absence of the target, while in reality, it is present. These hard negative and hard positive samples can effectively influence the learnable parameters when the network is trained to correct them.

The Siamese network learns the similarity function from the positive and negative image pairs during the training process. The proposed supervised hard sample mining method helps to mine both hard positive and hard negative samples to fine-tune the trained CFNet. The hard samples for learning are generated based on the influence factor derived from the online MOT results with the supervision from ground-truth trajectories. The approach we used to generate hard training samples is shown in Fig. 4. Based on the similarity score for data association, we assign a label, $z \in (-1, 1)$ to the image pairs, (x_L, d_i^f) that indicate whether the lost target, x_L is associated ($z = 1$) or not ($z = -1$) to the detections, D^f.

During the training process, an influence factor, $R_{\mathrm{IF}}(x_L, z)$ is evaluated using the ground-truth track and the decision from the data association module. To formulate the influence factor, we need another function variable $y(x_L) \in (-1, 1)$ whose value depends on the ground-truth trajectory corresponding to target, x_L.

$$y(x_L) = \begin{cases} 1, & \text{if } \max\left(overlap\_ratio(x_L^{GT}, d_i^f)\right) \geq T_o \\ -1, & \text{otherwise} \end{cases}, \quad (9)$$

where x_L^{GT} is the ground-truth track for the lost target and T_o is the overlap threshold. Equation (9) here means, $y = +1$ if there is a match for x_L in the given detections and $y = -1$ if there is no association possible, according to the ground truth.

Table 1 lists the assigned influence factor value for different decisions z on x_L by adopted the data association rule. It is inferred from the table that the influence factor, $R_{\mathrm{IF}} = +1$ whenever the data association took the right decision and influence factor, $R_{\mathrm{IF}} = -1$ if the decision is wrong. In the adopted data association system, CFNet is updated only when it makes a mistake in decision making. i.e. association estimation, z takes different actions as desired by the ground-truth trajectory. There are two instances for which the reward is negative. In the first case, the target is linked ($z = 1$) to an object detection d_m^f (refer (5)–(8)), which is incorrect according to the ground truth ($y = -1$). This indicates a false positive output from the trained CFNet. Then the image pair $\left(x_L, d_m^f\right)$ is added to the training database as a hard negative sample. In the second case, the decision is not to link ($z = -1$) with any of the detections from the detector. But the target, d_k^f is included in the given detections according to the ground truth ($y = 1$). That means, the association algorithm missed the right association. This indicates a false negative output from the trained CFNet. Then the image pair $\left(x_L, d_k^f\right)$ is included in the training database as a hard positive sample.

In the training process, we start with the CFNet model that was pre-trained on the ImageNet dataset. CFNet continues the learning process during the MOT using the training samples and updates the parameters based on the feedback from the dynamic status of the target. During training, the parameters of the CFNet model are updated by minimising the logistic loss over the new hard training samples, obtained from the supervised hard sample mining.

$$\arg\min_p \sum_j \mathscr{L}\left(\Psi_{w,\alpha,\beta}(x_j, d_j), y(x_j)\right), \quad (10)$$

where w, α and β are the learnable parameters, scale, and bias values, respectively. For each frame, we updated the parameters of the CFNet architecture. In the following frame, MOT is done with this updated CFNet. The trained CFNet model in the context of MOT performs better under the challenging conditions and provides better accuracy in tracking.

3.4 MOT algorithm

After accomplishing the training of Faster R-CNN with the MOT-17 dataset and CFNet using the supervised hard sample mining strategy, we utilise these trained architectures in the proposed MOT framework. The proposed MOT algorithm is summarised in Algorithm 1 given in Fig. 5. Given an image sequence, the goal of the MOT problem is to estimate the optimal sequential states of all the possible targets, i.e. the trajectory of each target. For each input frame, the Faster R-CNN detector outputs the person detections, D^f. Subsequently, in the data association part, all the targets under consideration in the tracked state get higher priority. The CFNet as a single-object tracker determines whether the target should stay in the tracked state or should be moved to the lost state. Then, for the lost targets, the CFNet acts as a similarity function to compute the pairwise similarity score with the object detections from the detector. Hungarian algorithm is then employed next to associate the detections to the lost targets based on the similarity score. The targets that are linked with the detections are reassigned as tracked targets. Finally, all the remaining detections that are not associated with any of the tracked targets are considered as new targets and a trajectory is initialised for each new detection. Here, to exempt the already assigned detection, non-maximum suppression based on bounding box overlap is applied. Nonetheless, as described in Section 1, to improve the detection performance, we incorporate feedback region proposals that provide temporal information about the previous trajectories, to the detector.

Table 1 Influence factor (R_{IF}) assigned for decision making in data association in lost state. $R_{\mathrm{IF}} = 1$ indicated the right decision and $R_{\mathrm{IF}} = -1$ indicated the wrong decision

$y(x_L)$	z	$R_{\mathrm{IF}}(x_L, z) = y(x_L) \times z$
1	1	1
−1	1	−1
1	−1	−1
−1	−1	1

4 Results

This section presents the experimental results of the proposed multiple object tracker on benchmark datasets focusing on person tracking to validate the efficiency and the tracking performance. To obtain comparable results with the state-of-the-art trackers, we evaluated our tracker framework on the MOT challenge dataset [29, 30], a standard reference when addressing MOT problems.

MOT challenge: MOT challenge dataset is a centralised benchmark dataset to test the MOT methods that include several categories of challenging tracking sequences with different characteristics such as object density, frame rate, occlusions, illuminations, etc. Mainly, there are three separate tracking sequence sets published by the MOT challenge, 2DMOT2015, MOT16, and MOT17. Each of the benchmark datasets includes separate video sequences for the training and the testing of the tracker. Training sequences are provided with public object detections from object detector and the ground-truth detections, whereas testing sequences only include object detections. The MOT17 dataset contains 14 challenging sequences of which seven are used for training and 7 for testing the tracker. The sequences are provided with three sets of detection from DPM [31], Faster R-CNN [8] and SDP [23] object detectors. The benchmark sequences included in MOT16 are the same as that of MOT17 with only DPM detection. The benchmark dataset 2DMOT2015 includes a total of 22 sequences each of which provided with ACF detections.

There are several metrics [32] for the quantitative evaluation of MOT that measure different aspects of tracker efficiency. The two important parameters in the MOT17 challenge that measure the object coverage and identity consistency are MOTA (MOT Accuracy) and IDF1 score. MOTP (MOT Precision) measures the misalignment between the groundtruth and the predicted bounding boxes. MT (mostly tracked) and ML (mostly lost) are another two parameters that indicate the percentage of ground-truth objects whose trajectories are covered by the tracking output. FP and FN represent, respectively, the total number of false positives and false negatives. Precision and recall are two derived metric, where precision is the fraction of true and relevant bounding boxes among the retrieved bounding boxes, while recall is the fraction of the total amount of relevant bounding boxes that were actually retrieved. The number of times a particular target changes its identity is measured by identity switches (IDSw). When the object is not detected in some frames, then fragmented trajectories are generated (Frag).

4.1 Implementation details

We accomplish the person detection using the ResNet-101-based Faster R-CNN model. By convention, we initialise the pre-trained model of Faster-RCNN and then retrained it with fine-tuning on the MOT17 dataset in order to improve its accuracy in the task

Input: Video Sequence as an ordered list of image frames, $V = \{I^f | f = 1, 2, \ldots, F\}$
Output: Set of object trajectories, $\mathcal{T} = \{\tau_i\}_{i=1}^{N}$, with $\tau_i = \{b_i^{f_s}, \ldots, b_i^{f_e}\}$, as a list of ordered target bounding boxes, where f_s and f_e are first and last frame in which target i exists, $b_i^{f_j} = (x, y, w, h)$

Initialization: $\mathcal{T} \leftarrow \emptyset$
Trained CNN models: Faster R-CNN with feedback region proposals retrained on MOT dataset as multiple object detector and CFNet trained using reward based supervised hard sample mining as data association metric.
for Video frame I^f in V **do**
  Person detections from Faster R-CNN, $D^f = \{d_j^f\}_{j=1}^{N_f}$
  if $(f == 1)$ **then**
    Initialize new trajectory τ_i^f for each detection,
    state==tracked;
  else
    for each tracked target $\tau_i^{(f-1)} \in \mathcal{T}$ **do**
      CFNet as single object tracker; find the new target location and similarity score, s_m.
      if $s_m < T_s$ **then**
        state==lost;
      end if
    end for
    for each lost targets, $\tau_i^{(f-1)} \in \mathcal{T}$ **do**
      CFNet as patch similarity function;
      for each detection d_k^f not covered by tracked objects **do**
        Obtain the similarity score map with lost target
        Compute the maximum similarity score, s_m
        if $s_m > T_s$ **then**
          detection considered for association with lost target
        end if
      end for
      Hungarian algorithm employed to assign detection to the lost targets.
      if lost $\tau_i^{(f-1)}$ assigned to detection d_m^f **then**
        state ==tracked;
      else
        if length of lost frames $> N_{inact}$ **then**
          state==inactive;
        end if
      end if
    end for
    for each detection d_j^f not covered by tracked and lost targets **do**
      Initialize a trajectory τ_i^f.
      state==tracked.
    end for
  end if
  for each $\tau_i^f \in \mathcal{T}$ **do**
    Predict the next location v using Lucas Kanade optical flow method.
    Feedback region proposals, R_{IF}=set of bounding boxes with center v and size same as τ_i.
    (Faster R-CNN classifier module cascade R along with its RPN module output).
  end for
end for
return set of trajectories of the objects, $\mathcal{T}$.

Fig. 5 *Algorithm 1: MOT algorithm*

domain. This fine-tuned Faster R-CNN is then integrated with the proposed MOT framework as a person detector. Similarly, the pre-trained CFNet architecture is trained within the context of MOT using training sequences of the MOT benchmark. The supervised hard sample mining strategy for adopted for training is detailed in Section 3.3. The learnable parameters w and correlation filter Ω_c are initialised from the pre-trained values of CFNet. We follow the back-propagation algorithm described in [11] to update the parameters of the network, by minimising the logistic loss over the new hard training samples. Training is conducted for 20 epochs for each new sample, with an initial learning rate of 0.01. For each frame, we updated the parameters of the CFNet architecture using the hard samples generated in the current frame.

In the proposed MOT framework, the decision for state transition of a target is based on two parameters, similarity threshold T_s and the maximum number of frames the target stays in the lost state before transferred into an inactive state, N_{inact}. Fig. 6 shows the IDF1 score and MT values with different values for similarity threshold, T_s. The value of T_s is a threshold value for the similarity score generated by the CFNet. If the similarity score is above the threshold T_s, the input image pairs to the CFNet are considered as similar. The optimum results obtained for the proposed tracker with the value of T_s equal to 0.4. In our analysis, we kept the value for N_{inact} as 20.

The proposed MOT algorithm is implemented in MATLAB with MatConvNet [33]. All experiments are conducted on a workstation with Intel Xeon X5675 at 3.06 GHz and an NVIDIA Geforce Titan Xp 12 GB GPU. The code is available at https://github.com/aswathyIIST/Feedback-Region-Proposals-for-MOT.

4.2 Analysis on validation dataset

In our MOT framework, the detector and tracker work simultaneously, hence are not considered as separate units. The proposed tracker generates its own object detections using the Faster R-CNN detector with the feedback region proposals. Therefore, the object detections provided in the MOT challenge database are not directly used in this analysis. Since the object detection annotations of the MOT test dataset are not released, we use the MOT training sequences to conduct analysis about our framework. The training data set is divided into training and validation sequences. The splitting of the sequences is shown in Table 2. We conducted our experiments to validate the importance of each contribution in the proposed MOT algorithm.

4.2.1 Ablation study: Contribution of different components: We investigate the contribution of different components in our framework by disabling one component at a time and then examining the performance drop in terms of MOTA on the validation set. Fig. 7 shows the significance of each component validating using the 2DMOT15 benchmark dataset. It is observed that the trend of the MOTA remains the same for the evaluation results on the MOT17 validation sequences. Table 3 presents the experimental results on all MOT evaluation metric, to demonstrate the importance of each component, evaluated on both MOT17 and 2DMOT2015 validation datasets.

In Fig. 7, the first set: feedback proposals, presents the importance of feedback region proposals from the tracker to the detector. It is evident that feedback improves the accuracy of the MOT framework. One of the challenging factors that affect the MOTA value in MOT evaluation is identity switches (IDSw), a measure that indicates the number of times the target changes its identity in the whole tracking process. Fragmentation is another factor that affects the performance of our proposed system. Fragmented trajectories are formed when identity switches do not occur, but the detector missed the target detection. The solution to these two problems is to reduce the number of missed detections. This can be done by improving the performance of the person detector. The feedback region proposals can be viewed as a reference given to the detector on the probable locations of existing targets. This helps the detector to reduce the missed detections and improve its efficiency. In the proposed full model MOT, we sent back the region proposal prediction for both tracked and lost targets. For the detailed analysis of this feedback proposal, we consider three different situations on feedback. (i) Only feedback tracked targets proposals (no lost targets), (ii) only feedback lost targets proposals and (iii) no feedback proposals. It is obvious from the results that with the feedback region proposals the performance of our tracking framework is improved. In addition to that, it is interested to note the accuracy difference with the region proposals with any one of the target states, tracked or lost. For the target in the tracked state, the appearance model of the tracked target is

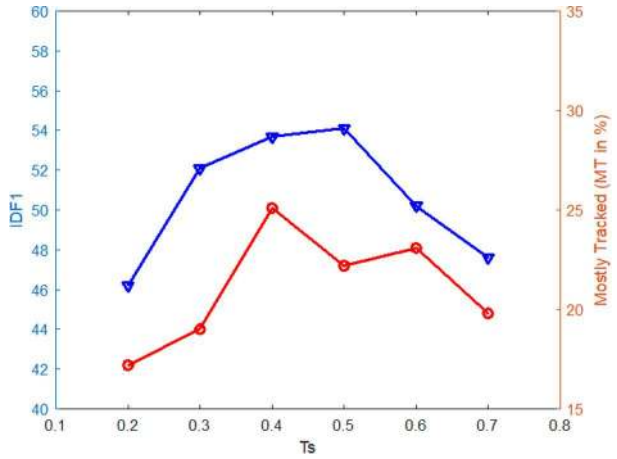

Fig. 6 *Performance analysis of the proposed MOT framework with different values for similarity threshold, T_s on 2DMOT2015 validation set. In the following analysis, T_s is set to 0.4*

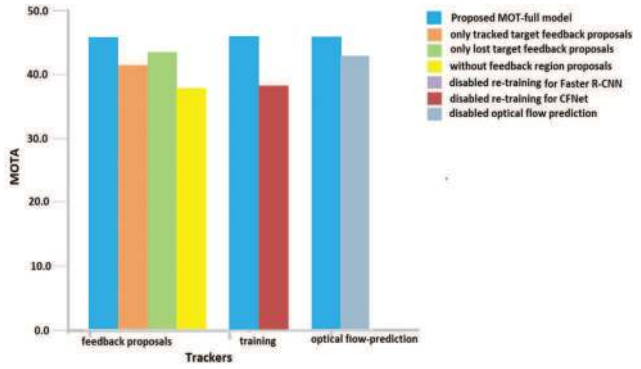

Fig. 7 *Analysis of the proposed MOT framework on the 2DMOT2015 validation sequences with different components. The complete MOT framework gives better results with the integration of each component*

Table 2 Training and validation sequences that are used to study the performance of the proposed MOT framework on the MOT benchmark

Training	Validation
2D MOT 15	
TUD-Stadtmitte	TUD-Campus
ETH-Bahnhof	ETH-Sunnyday
PETS09-S2L1	ETH-Pedcross2, Venice-2
ADL-Rundle-6, KITTI-13	ADL-Rundle-8, KITTI-17
MOT17	
MOT17-02	MOT17-04
MOT17-05	MOT17-09
MOT17-10	MOT17-11, MOT17-13

getting updated in each frame. Also, in the proposed MOT frame work, a single-object tracker is used to track the tracked targets. Therefore, chances are less for a tracked target to be in a lost category even if the detector missed the target detection. But in the case of lost targets, the position or the appearance model is not updated and the data association completely relies on the detections from the detector. Therefore, if the detector missed the correct target detection, the target will be continuing on its lost state and the tracking accuracy reduces.

In Fig. 7, the second set: training, indicates the significance of training on the pre-trained CNNs used in the proposed tracker. The Faster R-CNN object detector with ResNet-101 pre-trained on PASCAL-VOC and COCO training set is retrained on the MOT17 person detection dataset. As discussed in Section 3, the pre-trained CFNet is trained using the supervised hard sample mining supported by an influence factor for data association. It is clear from both Fig. 7 that the detector and tracker CNNs improve its accuracy after trained on the MOT benchmark dataset.

The third set: optical flow prediction, in Fig. 7 shows the relevance of the motion model of the target to predict the feedback region proposals. The motion model based on the Lucas–Kanade optical flow method with pyramids is employed here to predict the new location of the tracked or the lost targets. To study the contribution of the motion model, the current target location is fed back to the detector as the region proposals instead of the predicted location from the motion model. The results show an accuracy drop without optical flow-based motion prediction. Since the target in the tracked state is moving slowly from one frame to the next (usually verified from high frame rates), the regression module in the Faster R-CNN detector is able to refine the proposal bounding box of that slightly shifted target. But in case of lost targets, the position of the target is not updated for all the lost frames. Therefore, the regression module could not find the refined location of the particular region proposal. If the detector's own proposals do not include the lost target, then it causes a missed detection. By using the optical flow-based motion prediction, most of the time we could predict the location of the occluded targets and could avoid these missed detections and thus improve the tracking performance.

Table 3 shows the comparison of the proposed MOT framework with different variants of the proposed tracker by disabling different components, on the MOT17 and 2DMOT2015 validation datasets. The best values for this evaluation are given in **boldface** in Table 3. It is evident from the results that each component significantly contributes to the improvement of the performance of the proposed MOT framework. It is clear from Table 3 that the MT, ML, IDSw and fragmentation metrics improved with feedback region proposals. The proposed feedback region proposals help to improve the detection accuracy and reduce the missed detections. This helps to improve the overall performance of the MOT system. In the proposed MOT framework we incorporate two pre-trained CNN architectures after training on the MOT benchmark dataset. It is inferred from Table 3 that the trained detector and tracker CNNs help to enhance the performance of the MOT tracker. The prediction of probable locations of the existing targets is predicted

Table 3 Analysis of the proposed MOT framework on the validation datasets, both MOT17 and 2DMOT2015, and comparison with different proposed tracker variants by disabling different components

Tracker	MOTA ↑	MOTP ↑	IDF1 ↑	MT ↑	ML ↓	FP ↓	FN ↓	Recall ↑	Precision ↑	IDSw ↓	Frag.↓
MOT17											
proposed MOT	**60.8**	**78.2**	**53.7**	**74(25.1)**	**41(14.1)**	**4899**	**23,758**	**62.1**	**88.8**	**312**	**728**
proposed MOT without feedback proposals	57.2	76.7	51.3	57(19.3)	61(20.8)	5936	25,158	58.7	85.7	506	851
proposed MOT without training	57.2	77.3	51.5	67(22.8)	56(19.2)	5381	25,693	59.3	87.4	478	798
proposed MOT without optical flow prediction	60.2	77.9	52.8	70(23.8)	44(14.8)	5039	23,978	61.5	88.4	352	749
2DMOT2015											
proposed MOT	**46.2**	**75.3**	**47.6**	**52(22.2)**	**36(15.4)**	**2135**	**10,158**	**55.4**	**85.5**	**124**	**256**
proposed MOT without feedback proposals	38.2	71.9	46.1	42(18.0)	47(20.0)	3092	10,956	52.0	79.3	218	321
proposed MOT without training	38.1	73.1	46.7	45(19.2)	51(21.8)	2854	11,348	53.6	82.1	186	298
proposed MOT without optical flow prediction	43.2	74.2	47.1	49(21.1)	38(16.4)	2659	10,293	54.8	82.4	143	261

(The best values are in **boldface**).

using optical flow-based motion model. The optical flow prediction refines the location of the feedback region proposals and thereby improves the MOT tracker framework.

4.2.2 Performance analysis with Siamese CNN variants: The data association method designed in the proposed MOT framework exploits the power of Siamese CNN as a single-object tracker and a similarity metric. To study the performance of the proposed tracker with different Siamese CNN structures, we employed some of the existing variants of the Siamese network: SiameseFC [10], CFNet [11] and DCFNet [34], for data association. The experimental results on MOT17 validation data sequences are given in Table 4. We also compared the three variants of CFNet: CFNet-conv5 with five convolutional layers, CFNet-conv2 with two convolutional layers and CFNet-conv2(triplet) with two convolutional layers and uses triplet loss [12] for training.

From the experimental results obtained, it is observed that the performance of the proposed MOT framework is directly related to the performance of the Siamese network as a single-object tracker. Generally, if we deploy a Siamese CNN that achieves better speed and accuracy, the performance of the proposed MOT framework also will get improved. Among the variants of Siamese CNN tested here, the CFNet with two convolutional layers (CFNet-conv2) obtains high speed and slightly lower performance than the best. Thus, it is selected as the data association architecture in our proposed MOT framework.

4.3 Evaluation on test dataset

The proposed MOT framework is evaluated on the MOT test dataset (on both MOT17 and 2DMOT2015). Some of the recent and better performing multiple object trackers are selected for the comparison study with our approach. These state-of-the-art trackers evaluated with public detections provided by the MOT challenge. For a fair comparison with these trackers, the proposed MOT framework is slightly modified to perform all the test data evaluation with the given public MOT detections. Here, we are not using our Faster-RCNN detector to find new detections and all the new tracks are initialised only from the frame to frame detections provided with the MOT dataset. To deal with the proposed feedback region proposals, only the classifier and the regression modules of the Faster R-CNN are used. The classifier assigns a class score for each proposal and the regression module realigns the bounding box to fit with the object. From the bounding boxes, the person detections are selected based on the class scores. The detections generated here can be considered as private detections. We forward the MOT detections along with the private detections computed from the feedback region proposals to the MOT tracker part. To avoid multiple detection entries for the same object, a non-maximum suppression that based on bounding box overlap is employed before the tracker module.

The proposed multiple object tracker trained on MOT training sequences is then tested on the MOT17 and 2DMOT2015 testing datasets. Our experimental results are then submitted to the MOT challenge website for evaluation. Table 5 summarises the tracking performance of our proposed tracker on the MOT benchmark, where we compared it with other tracking methods. Moreover, the contributions and impacts of various components, such as feedback region proposals, supervised hard sample mining strategy for learning and optical flow-based motion prediction model are also given in Table 5. It is clear from the results that the introduction of the feedback region proposals model helps to reduce the missed detections and improved the performance of the proposed tracker. It is also inferred from Table 5 that the re-trained CNN models and

Table 4 Analysis of the proposed MOT framework with variants of Siamese CNN on the MOT17 validation dataset

Tracker	MOTA ↑	MOTP ↑	IDF1 ↑	MT ↑	ML ↓	FP ↓	FN ↓	Recall ↑	Precision ↑	IDSw ↓	Frag.↓	Hz ↑
proposed MOT + SiamFc	55.3	73.2	46.7	17.1	21.2	6479	25,928	58.3	84.8	594	874	**2.0**
proposed MOT + CFNet-conv5	60.4	75.3	48.3	23.5	15.1	5132	23,812	61.3	88.0	**309**	725	1.4
proposed MOT + CFNet-conv2	60.8	*78.2*	53.7	*25.0*	*14.1*	4899	23,758	*62.1*	*88.8*	*312*	*712*	*1.8*
proposed MOT + CFNet-conv2 (triplet)	*61.1*	77.8	*54.1*	**26.0**	14.8	*4758*	*23,663*	60.1	88.2	328	**698**	1.7
proposed MOT + DCFNet	**61.6**	**78.9**	**54.5**	**26.0**	**13.1**	**4623**	**23,398**	**62.8**	**89.5**	341	731	1.5

(The best results are in bold and the second best in italic).

Table 5 Comparison of the proposed MOT framework on the test dataset, both MOT17 and 2DMOT2015, with state-of-the-art trackers

Tracker	MOTA ↑	MOTP ↑	IDF1 ↑	MT ↑	ML ↓	FP ↓	FN ↓	Recall ↑	Precision ↑	IDSw ↓	Frag.↓
MOT17											
proposed MOT	***53.2***	**78.2**	53.8	*23.5*	**29.8**	*13,619*	248,529	**56.2**	*95.8*	**1649**	*3781*
proposed MOT without feedback proposals	51.3	76.4	51.2	18.7	34.2	14,316	257,208	52.1	95.1	2941	4279
proposed MOT without training	52.9	77.3	51.8	19.2	***32.8***	13,983	249,431	54.2	95.5	1982	3974
proposed MOT without optical flow prediction	53.1	***77.9***	52.5	***21.8***	*30.7*	13,720	248,921	***56.0***	***95.7***	*1728*	***3842***
tractor++ [18]	*53.5*	*78.0*	52.3	19.5	36.6	**12,201**	***248,047***	***56.0***	**96.3**	2072	4611
DMAN [17]	48.2	75.7	*55.7*	19.3	38.3	26,218	263,608	53.3	92.0	2194	5378
Siamese Track-RCNN [35]	**59.6**	NA	**60.1**	**23.9**	33.9	15,532	**210,519**	NA	NA	2068	NA
DEEPTAMA [36]	50.3	76.7	53.5	19.2	37.5	25,479	252,996	55.2	92.4	2192	3978
FAMNet [37]	52.0	76.5	48.7	19.1	33.4	***14,138***	253,616	55.1	95.6	3078	5318
jCC [38]	51.2	75.9	***54.5***	20.9	37.0	25,937	*247,822*	*56.1*	92.4	***1802***	**2984**
2DMOT2015											
proposed MOT	*44.3*	***74.6***	***46.8***	*21.7*	*26.2*	*5942*	***27,312***	*56.3*	**85.5**	***932***	***1473***
proposed MOT without feedback proposals	37.0	72.1	44.5	17.0	30.4	7321	29,917	52.0	81.5	1429	1928
proposed MOT without training	41.3	72.9	45.3	19.1	29.3	6842	27,932	54.7	83.1	1242	1956
proposed MOT without optical flow prediction	43.2	73.0	45.9	***20.9***	27.1	***6314***	27,532	***55.7***	*84.5*	1023	1732
tracktor++ [18]	***44.1***	*75.0*	46.7	18.0	*26.2*	6477	*26,577*	**56.7**	84.3	1318	1790
FFT [39]	**46.3**	**75.5**	**48.8**	**29.1**	**23.2**	9870	**21,913**	NA	NA	1232	1638
KCF [40]	38.9	70.6	44.5	16.6	31.5	7321	29,501	52.0	81.4	*720*	**1440**
AMIR15 [41]	37.6	71.7	46.0	15.8	***26.8***	7933	29,397	52.2	80.2	1026	2020
AM [42]	34.3	70.5	*48.3*	11.4	43.4	**5154**	34,846	43.3	***83.8***	**348**	*1463*

(Bold for the best values, italic for the second place and bold italic for the third place). NA represents the values that are not available in the publications.

region proposal prediction using the optical flow-based motion model incorporates in the proposed MOT framework enhances the tracking performance. The experiment results show that in comparison with other MOT methods, the proposed MOT framework achieves competitive tracking performance. The evaluation on the benchmark datasets ensures that the proposed MOT framework incorporates an improved object detection followed by enhanced data association and tracking methods. From the experimental results, it is inferred that the missed detections and false alarms are significantly reduced in the proposed MOT framework.

5 Conclusion

In this study, we developed a unified MOT framework with an efficient object detection module and an accurate data association method. The Faster R-CNN person detector with feedback region proposals from the tracker reduces the missed detections and provides better object detections that in turn help to improve the data association accuracy. The data association algorithm designed here exploits the strengths of the correlation filter-based Siamese CNN (CFNet) as a single-object tracker and a similarity metric. Furthermore, we proposed a supervised hard sample mining strategy supported by an influence factor, derived from the online tracking results, to train the Siamese network. The complete MOT system is trained and evaluated over the benchmark MOT challenge datasets. When comparing with the state-of-the-art trackers, it is observed that the proposed MOT algorithm performs better in terms of MOT evaluation metrics. The evaluation results on the validation dataset also show the relevance of each proposed component in the MOT framework. Moreover, from the experimental results obtained, it is observed that the performance of the proposed MOT framework is directly related to the performance of the Siamese network employed. Therefore, with a Siamese CNN with better accuracy and speed, the performance of the proposed MOT framework can be further improved.

6 References

[1] Zhang, J., Presti, L., Sclaroff, S.: 'Online multi-person tracking by tracker hierarchy'. Proc. IEEE Int. Conf. Advanced Video Signal-Based Surveillance, Beijing, China, September 2012, pp. 379–385

[2] Yoon, J.H., Lee, C.R., Yang, M.H., *et al.*: 'Online multi-object tracking via structural constraint event aggregation'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, Las Vegas, NV, USA, June 2016, pp. 1392–1400

[3] Chen, L., Peng, X., Ren, M.: 'Recurrent metric networks and batch multiple hypothesis for multi-object tracking', *IEEE Access*, 2018, **7**, pp. 3093–3105

[4] Henriques, J.F., Caseiro, R., Batista, J.: 'Globally optimal solution to multi-object tracking with merged measurements'. Proc. IEEE Int. Conf. Computer Vision, Barcelona, Spain, November 2011, pp. 2470–2477

[5] Milan, A., Roth, S., Schindler, K.: 'Continuous energy minimization for multitarget tracking', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2014, **36**, (1), pp. 58–72

[6] He, K., Zhang, X., Ren, S., *et al.*: 'Spatial pyramid pooling in deep convolutional networks for visual recognition', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2015, **37**, (9), pp. 1904–1916

[7] Girshick, R.: 'Fast R-CNN'. IEEE Int. Conf. on Computer Vision (ICCV), Santiago, Chile, December 2015

[8] Ren, S., He, K., Girshick, R., *et al.*: 'Faster R-CNN: towards real-time object detection with region proposal networks'. Advances in Neural Information Processing Systems, Montreal, Quebec, Canada, 2015, pp. 91–99

[9] Redmon, J., Divvala, S., Girshick, R., *et al.*: 'You only look once: unified, real-time object detection'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, Caesars Palace, Las Vegas, Nevada, United States, 2016, pp. 779–788

[10] Bertinetto, L., Valmadre, J., Henriques, J.F., *et al.*: 'Fully-convolutional Siamese networks for object tracking'. European Conf. on Computer Vision, Amsterdam, The Netherlands, 2016, pp. 850–865

[11] Valmadre, J., Bertinetto, L., Henriques, J.F., *et al.*: 'End-to-end representation learning for correlation filter based tracking'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, Honolulu, HI, USA, July 2017, pp. 5000–5008

[12] Dong, X., Shen, J.: 'Triplet loss in Siamese network for object tracking'. Proc. European Conf. on Computer Vision (ECCV), Munich, Germany, 2018, pp. 459–474

[13] Li, B., Yan, J., Wu, W., *et al.*: 'High performance visual tracking with Siamese region proposal network'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, June 2018, pp. 8971–8980

[14] Wen, L., Li, W., Yan, J., *et al.*: 'Multiple target tracking based on undirected hierarchical relation hypergraph'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, Columbus, OH, USA, June 2014, pp. 1282–1289

[15] Kim, C., Li, F., Ciptadi, A., *et al.*: 'Multiple hypothesis tracking revisited'. Int. Conf. on Computer Vision, Santiago, Chile, December 2015, pp. 4696–4704

[16] Sun, S., Akhtar, N., Song, H., *et al.*: 'Deep affinity network for multiple object tracking', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019, pp. 1–1

[17] Zhu, J., Yang, H., Liu, N., *et al.*: 'Online multi-object tracking with dual matching attention networks'. Int. Conf. on Computer Vision, Munich, Germany, 2018

[18] Bergmann, P., Meinhardt, T., Leal-Taixe, L.: 'Tracking without bells and whistles'. Proc. of the IEEE Int. Conf. on Computer Vision, Seoul, South Korea, 2019

[19] Lan, L., Wang, X., Hua, G., *et al.*: 'Semi-online multi-people tracking by re-identification', *Int. J. Comput. Vis.*, 2020, **128**, pp. 1937–1955

[20] Lan, L., Wang, X., Zhang, S., *et al.*: 'Interacting tracklets for multi-object tracking', *IEEE Trans. Image Process.*, 2018, **27**, (9), pp. 4585–4597

[21] Uijlings, J.R.R., Van De Sande, K.E.A., Gevers, T., *et al.*: 'Selective search for object recognition', *Int. J. Comput. Vis.*, 2013, **104**, (2), pp. 154–171

[22] Girshick, R., Donahue, J., Darrell, T., *et al.*: 'Rich feature hierarchies for accurate object detection and semantic segmentation'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, Columbus, OH, USA, June 2014, pp. 580–587

[23] Yang, F., Choi, W., Lin, Y.: 'Exploit all the layers: fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, Las Vegas, NV, USA, June 2016, pp. 2129–2137

[24] Guo, Q., Feng, W., Zhou, C., *et al.*: 'Learning dynamic Siamese network for visual object tracking'. Proc. IEEE Int. Conf. Computer Vision, Venice, Italy, October 2017, pp. 1781–1789

[25] Tao, R., Gavves, E., Smeulders, A.W.: 'Siamese instance search for tracking'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, Las Vegas, NV, USA, June 2016, pp. 1420–1429

[26] He, A., Luo, C., Tian, X., *et al.*: 'A twofold siamese network for real-time object tracking'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, June 2018, pp. 4834–4843

[27] Bouguet, J.Y: 'Pyramidal implementation of the affine Lucas Kanade feature tracker description of the algorithm', *Intel Corp.*, 2001, **5**, pp. 1–10

[28] Xiang, Y., Alahi, A., Savarese, S.: 'Learning to track: online multi-object tracking by decision making'. Int. Conf. on Computer Vision, Santiago, Chile, December 2015, pp. 4705–4713

[29] Leal-Taixe, L., Milan, A., Reid, I., *et al.*: 'MOTChallenge 2015: towards a benchmark for multi-target tracking', arXiv:1504.01942 [cs], 2015

[30] Milan, A., Leal-Taixe, L., Reid, I., *et al.*: 'MOT16: a benchmark for multi-object tracking', arXiv:1603.00831 [cs], 2016

[31] Felzenszwalb, P.F., Girshick, R.B., McAllester, D., *et al.*: 'Object detection with discriminatively trained part-based models', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2010, **32**, pp. 1627–1645

[32] Bernardin, K., Stiefelhagen, R.: 'Evaluating multiple object tracking performance: the CLEAR MOT metrics', *Image Video Process.*, 2008, **2008**, (1), pp. 1–10

[33] Vedaldi, A., Lenc, K.: 'Matconvnet: convolutional neural networks for MATLAB'. ACM Int. Conf. on Multimedia, Brisbane, Australia, 2015

[34] Wang, Q., Gao, J., Xing, J., *et al.*: 'DCFNet: discriminant correlation filters network for visual tracking', arXiv preprint arXiv:1704.04057, 2017

[35] Shuai, B., Berneshawi, A.G., Modolo, D., *et al.*: 'Multi-object tracking with Siamese track-RCNN', arXiv:2004.07786, April 2020

[36] Yoon, Y., Kim, D., Yoon, K., *et al.*: 'Online multiple pedestrian tracking using deep temporal appearance matching association', arXiv:1907.00831, 2019

[37] Chu, P., Ling, H.: 'FAMNet: joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking'. Int. Conf. on Computer Vision, Seoul, S. Korea, 2019

[38] Keuper, M., Tang, S., Andres, B.: 'Motion segmentation and multiple object tracking by correlation co-clustering', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020, **42**, (1), pp. 140–153, doi: 10.1109/TPAMI.2018.2876253

[39] Zhang, J., Zhou, S., Chang, X., *et al.*: 'Multiple object tracking by flowing and fusing', arXiv:2001.11180, January 2020

[40] Chu, P., Fan, H., Tan, C., *et al.*: 'Online multi-object tracking with instance-aware tracker and dynamic model refreshment'. Winter Conf. on Applications of Computer Vision, Waikoloa Village, HI, USA, January 2019, pp. 161–170

[41] Sadeghian, A., Alahi, A., Savarese, S.: 'Tracking the untrackable: learning to track multiple cues with long-term dependencies'. Int. Conf. on Computer Vision, Venice, Italy, October 2017, pp. 300–311

[42] Chu, Q., Ouyang, W., Li, H., *et al.*: 'Online multi-object tracking using CNN-based single object tracker with spatial-temporal attention mechanism'. Int. Conf. on Computer Vision, Venice, Italy, October 2017, pp. 4846–4855