Graphical Abstract

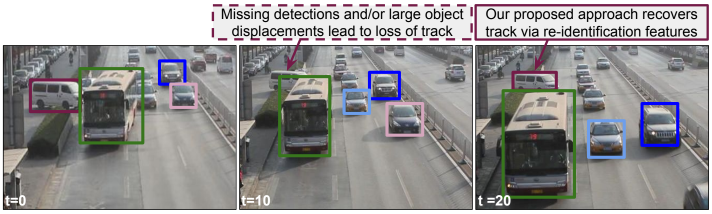
Joint Detection and Tracking in Videos with Identification Features

Bharti Munjal, Abdul Aftab Rafey, Sikandar Amin, Meltem Demirkus, Federico Tombari, Fabio Galasso

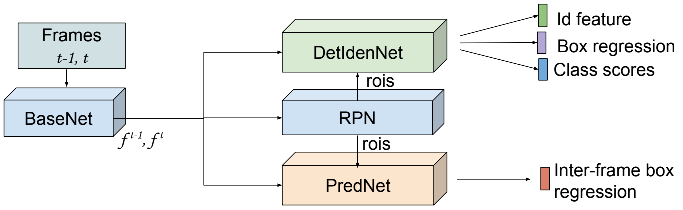
ELSEVIER

Recent work has shown that combining object detection and tracking tasks, in the case of video data, results in a considerably higher performance for both tasks, especially when a joint formulation is used. However, typical approaches for such a joint formulation require a relatively high video frame rate as they inherently rely on the overlap between detection and prediction bounding boxes for object association.

By contrast low frame rates, fast moving objects, and/or missing detections (e.g. see the illustration below), cause large object displacements, in which cases recent approaches resort to re-identification features for the association.


Missing detections and/or large object displacements lead to loss of track
Our proposed approach recovers track via re-identification features
t=0
t=10
t =20

Here we propose a joint optimization of detection, tracking and identification features for videos, via a single end-to-end neural network model, depicted below. To our knowledge, this is the first joint formulation of the re-identification task in the context of tracking, which maintains the detector performance.



Our method reaches the state-of-the-art on MOT and UA-DETRAC datasets. It ranked 1st among the online trackers in the UA-DETRAC'18 tracking challenge, and 3rd overall.

Research Highlights

- We formulate detection, tracking and ID feature learning into an end-to-end joint neural network model for the first time.
- We demonstrate that our model is on par with the state-of-the-art methods addressing detection and tracking separately.
- We demonstrate that our model outperforms the state-of-the-art methods at low frame rates.
- We ranked 3rd in the UA DETRAC '18 tracking challenge and 1st among the online trackers.
- We achieve the state-of-the-art performance on the MOT dataset.

Pattern Recognition Letters
journal homepage: www.elsevier.com

Joint Detection and Tracking in Videos with Identification Features

Bharti Munjal[a,b,**], Abdul Aftab Rafey[a], Sikandar Amin[a], Meltem Demirkus[a], Federico Tombari[b], Fabio Galasso[a]

[a]*OSRAM, Parkring 33 Garching, Munich 85748, Germany*
[b]*Technische Universität München, Boltzmannstraße 3 Garching, Munich 85748, Germany*

ABSTRACT

Recent works have shown that combining object detection and tracking tasks, in the case of video data, results in a considerably higher performance for both tasks, especially when a joint formulation is used. Typical approaches for such a joint formulation require a relatively high video frame rate as they inherently rely on the overlap between detection and prediction bounding boxes for object association. In the case of large object displacements, such as with low frame rates (or fast moving objects), recent approaches resort instead to object identification features for the association.

In this work, we propose the joint optimization of detection, tracking and identification features for videos. To our knowledge, we are the first to formulate re-identification jointly in the context of tracking. Moreover, we show that such a joint optimization also maintains the detector performance. Using the detections, predictions of the bounding boxes in the next frames, and corresponding identification features within a simple tracking framework addresses large motion displacements across frames, which we demonstrate on the low-frame-rate videos. Moreover, the proposed method reaches the state-of-the-art on MOT and UA-DETRAC datasets. Specifically, in the UA-DETRAC'18 tracking challenge, our method ranked 1st among the online trackers and 3rd overall.

1. Introduction

Object detection and recognition are long standing challenges in computer vision (Benenson et al., 2014), being essential requirements for applications such as scene understanding, video-surveillance and robotics. Of equal importance is tracking, which is often necessary to deal with dynamic scenes in the aforementioned application scenarios (Milan et al., 2016). Person re-identification, i.e. associating a person's identity across different viewpoints, is a relatively recent task in computer vision, although it leverages relevant literature on image retrieval (Almazán et al., 2018).

Recently, detection in videos has emerged as a challenge (Real et al., 2017). Intuitively, processing videos enhances objects which move, and accumulating evidence over time makes detection more robust (Zhu et al., 2018b). With similar arguments, (Feichtenhofer et al., 2017) has recently shown that jointly addressing detection and tracking improves detection. Interestingly, however, re-identification has been so far researched for images and, somehow surprisingly, a joint formulation of re-identification and detection degrades the detector performance.

In this paper, we consider for the first time detection, tracking and re-identification altogether, formulating an end-to-end joint neural network which detects people, provides tracking associations across and estimates id-features, to match people across frames further apart. Our contribution includes extending re-identification to videos, demonstrating that, when applied to video, re-identification does not harm the joint detection nor tracking task. Our proposed approach requires training and inference of a single feed-forward joint model.

In addition to analyzing detection performance, we integrate the model into a simple IoU-based tracker and compare it to the state-of-the-art on the challenging MOT (Milan et al., 2016) and UA-DETRAC (Lyu et al., 2017; Wen et al., 2015) database. More in details, our detections are matched over time by using the IoU between detections and tracked bounding boxes, and selecting optimal matches with the Hungarian algorithm. The re-identification branch of our model provides ID-features, which additionally weight the IoU-based matches. Our results

[**] Corresponding author: M.Bharti@osram.com (Bharti Munjal)

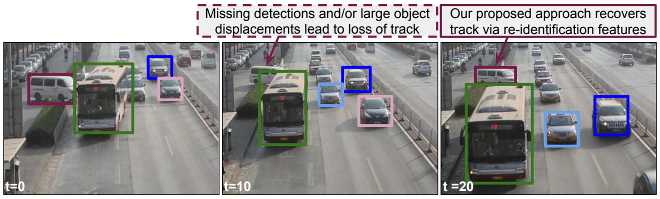

Missing detections and/or large object displacements lead to loss of track
Our proposed approach recovers track via re-identification features
t=0
t=10
t =20

Fig. 1. Our proposed approach maintains the ID of the tracked objects in difficult cases such as missing detections and/or large displacements, given by fast motion or low video frame rates. This is accomplished by the joint optimization of detection, tracking and re-identification, within a single end-to-end model.

are state-of-the-art on videos with high frame-rate, but clearly surpass it for lower frame-rate videos. On the UA-DETRAC'18 challenge, we rank 3^{rd} overall and 1^{st} among online trackers who attended the challenge. We show 11.6 points improvement in PR-MOTA over last year's winning tracker. Furthermore, we report state-of-the-art performance of the proposed framework on the MOT 2016 (Milan et al., 2016) benchmark.

Our main contributions are: *(i)* We formulate detection, tracking and identification feature learning into an end-to-end joint neural network model for the first time; *(ii)* we demonstrate that the joint model is on par with state-of-the-art methods addressing detection and tracking separately, but it outperforms them at low frame rates, where re-identification provides guidance to matching; and *(iii)* we ranked 3^{rd} in the UA-DETRAC'18 tracking challenge and 1^{st} among the online trackers, and we achieve state-of-the-art performance on the MOT dataset.

2. Related Work

2.1. Object detection and tracking

In the object detection field, the best detectors are either region based or single shot ones. Region based detector was first proposed in (Girshick et al., 2014) and subsequently improved over the past few years with the works of Fast-RCNN (Girshick, 2015), Faster-RCNN (Ren et al., 2015) and RFCN (Jifeng Dai and Sun, 2016). These methods use a set of object proposals generated by a Region Proposal Network (Girshick, 2015), followed by a neural network based classification and regression. The second family of object detectors skips the proposal generation step and directly regresses the bounding boxes and respective class scores from an image. The two popular work in this category are YOLO (Joseph Redmon, 2016) and SSD (Liu et al., 2016).

Recently, detection in videos has been receiving a lot of attention due to emerging challenges and databases, e.g. (Real et al., 2017; Russakovsky et al., 2015). Consequently, many recent works focus on improving detection performance for the case of videos. As shown in (Feichtenhofer et al., 2017; Zhu et al., 2018b), use of accumulated object evidence over time for a given video increases detection performance. For instance, (Feichtenhofer et al., 2017) reports state-of-the-art detection performance via combining detection losses from 2 consecutive frames with a tracking loss that is obtained from an additional correlation layer.

The object tracking literature contains a number of challenging benchmarks, such as for the single object tracking case: VOT (Kristan et al., 2017) and OTB (Wu et al., 2015), and for the multi-object tracking (MOT) case: UA-DETRAC (Wen et al., 2015), MOT (Milan et al., 2016). The single object tracking community mostly focuses on online object tracking, where a tracker is initialized with a ground-truth bounding box and continues tracking until it is observed that it drifts away, and only then it is re-initialized. On the other hand, multi-object tracking community focuses on both online and offline approaches, and uses the tracking-by-detection centered mentality, where the tracker is regularly initialized by the detector even if there is no need to. The winner of UA-DETRAC challenge, for instance, is an offline IoU tracker: (Bochinski et al., 2017), that creates multiple trajectory hypothesis using IoU metric on detection bounding boxes.

2.2. Re-Identification

The re-identification literature focuses on the problem of matching a given target person to a set of gallery images observed in different cameras. This means learning a unique feature vector for each identity, which should be invariant to the changes in camera viewpoint, pose, illumination, occlusion and background clutter. Most work in the re-identification field use cropped bounding boxes of objects, e.g. (Kalayeh et al., 2018; Chen et al., 2017). It is shown in (Xiao et al., 2017) that one can utilize a joint formulation of re-identification and detection on full images, and still be on par with the state-of-the-art. Furthermore, such identity features (ID-features) can be adapted for tracking problem, especially given advantages of the use of methods like (Xiao et al., 2017). However, our experiments show that such a joint formulation degrades the detector performance which is critical for a good tracking performance.

3. Methodology

We address object detection and tracking by jointly training for the three tasks of detection, prediction, and re-identification. In contrast to traditional sequential approaches for detection and tracking, our focus is on proposing a unified end-to-end architecture which optimizes these three objectives together. In this section, we first give an overview of our approach and details of the two main branches of our model architecture, i.e. *DetIdenNet* and *PredNet*. Then we describe our multi-task objective. Finally, we discuss our simple tracking algorithm, which uses the detections and identification features by *DetIdenNet* and the corresponding predictions in the next frame by *PredNet* to generate robust trajectories of multiple objects.

3.1. Model Architecture

Our proposed model architecture is given in Figure 2 with two output branches *DetIdenNet* and *PredNet*. The architecture takes two consecutive frames $I^{t-\delta}$ and $I^t \in^{H\times W\times 3}$ at times $t-\delta$ and t of a video sequence. The two frames are first passed through *BaseNet*, composed of the first four blocks ($Conv1-4$) of a *ResNet* architecture (He et al., 2016). $f^{t-\delta}$ and $f^t \in^{h\times w\times d}$, represent the base features for the two input images, where h, w and d are the height, width and number of channels of base features respectively. On top of these features, we employ a region proposal network (RPN) (Ren et al., 2015) to get independent object proposals from both frames, which are then forwarded to the *DetIdenNet* branch.

3.1.1. Detection and Identification network

For each proposal from the RPN, *DetIdenNet* pools features of size 7×7 from the respective base features using ROI-Align (Kaiming He and Girshick, 2017). The pooled features are then passed through a set of convolutions composed of the last convolutional block (*Conv*5) of *ResNet* (He et al., 2016) followed by a global average pooling, giving 2048 dimensional feature for each proposal. This feature is then passed to two sibling branches for identification and detection respectively. The identification branch first applies a fully connected layer of size 256, giving a lower dimensional identity feature for each proposal as suggested by Xiao (Xiao et al., 2017). The detection branch applies two parallel fully connected layers of size $(C + 1)$ and $4 \times (C + 1)$ giving class scores as well as class specific bounding box regression for each proposal, where $(C + 1)$ are the number of classes including background. During training, the detection branch employs Softmax Cross Entropy loss for classification and Smooth L1-loss for bounding box regression as in (Ren et al., 2015). Following (Xiao et al., 2017), we use the Online Instance Matching Loss (*OIM*) to learn the 256 dimensional identity feature embedding.

3.1.2. Prediction network

Given the base features $f^{t-\delta}$ and f^{t} of size $h \times w \times d$ of the two input images, *PredNet* aims to predict the regression targets for the object detections from the first image to the second one. To achieve this, *PredNet* first applies a Correlation Layer that finds the correlation of each feature $f^{t-\delta}(x, y)$ in the first feature map with its neighbouring $(2n + 1) \times (2n + 1)$ window in the second feature map as suggested by Feichtenhofer (Feichtenhofer et al., 2017). This produces feature map C of size $h(2n + 1) \times w(2n + 1)$ as shown in Eq. 1.

$$C(x, y) = \sum_{d} f^{t-\delta}(x, y, d) 1(n, n) \odot N[f^{t}(x, y, d)] \qquad (1)$$

where $f^{t-\delta}(x, y, d)$ and $f^{t}(x, y, d)$ are scalar values at spatial position x, y and channel d in feature maps $f^{t-\delta}$ and f^{t} respectively. $1(n, n)$ is a $(2n + 1) \times (2n + 1)$ matrix of ones used to repeat the scalar value $f^{t-\delta}(x, y, d)$ to a $(2n + 1) \times (2n + 1)$ matrix. $N[f^{t}(x, y, d)]$ is $(2n + 1) \times (2n + 1)$ neighbourhood matrix of $f^{t}(x, y, d)$. The above equation first computes the element-wise multiplication $\odot$ of the matrix $f^{t-\delta}(x, y, d)1(n, n)$ with the neighbourhood matrix $N[f^{t}(x, y, d)]$ and then sum it along the channel dimension. It should be noted that $C(x, y)$ is a block of size $(2n + 1) \times (2n + 1)$ giving the correlation of feature $f^{t-\delta}(x, y)$ with $(2n + 1) \times (2n + 1)$ neighbourhood in $f^{t}(x, y)$. The correlated feature map C is then passed to a Patch Summary layer that summarizes each $(2n + 1) \times (2n + 1)$ window using a convolution of filter size $(2n+1) \times (2n+1)$, stride $2n+1$ and output channels 512. This summarized feature map of size $(h \times w \times 512)$ is then concatenated with original base features $f^{t-\delta}$ and f^{t} followed by a 1×1 convolution to obtain d output channels which allows to employ standard *Conv*5 block of the *ResNet* architecture. These features are then passed to ROI-Align (Kaiming He and Girshick, 2017) together with the detection bounding boxes of the first frame (track rois), followed by *Conv*5 block of *ResNet*, and a fully connected layer to give the regression, $\Delta^{t} = (\Delta^{t}_{x}, \Delta^{t}_{y}, \Delta^{t}_{w}, \Delta^{t}_{h})$ for each track roi. During training, *PredNet* uses a Smooth L1 loss between ground-truth targets and predicted targets as in (Feichtenhofer et al., 2017).

3.1.3. Multi-task objective.

To train the branches of our network jointly, we use a multi-task loss that combines the classification loss (L_{cls}), regression loss (L_{reg}), together with inter-frame bounding box regression loss (L_{tra}) and identification loss (L_{iden}). The overall loss for a batch of N rois is given as,

$$\begin{aligned} L = \frac{1}{N} \sum_{j=1}^{N} L_{cls}(p_{j,c^*}) + \lambda_1 \frac{1}{N_{fg}} \sum_{j=1}^{N} [c^*_j > 0] L_{reg}(b_j, b^*_j) \\ + \lambda_2 \frac{1}{N_{tra}} \sum_{j=1}^{N_{tra}} L_{tra}(\Delta^{t}_{j}, \Delta^{*,t}_{j}) + \lambda_3 \frac{1}{N_{iden}} \sum_{j=1}^{N_{iden}} L_{iden}(q_{j,i^*}) \end{aligned} \qquad (2)$$

where for each roi j, p_{j,c^*} is the predicted classification probability of its ground-truth class c^* and b_j, b^*_j are the predicted and ground-truth bounding box targets. In the tracking loss, Δ^{t}_{j} and $\Delta^{*,t}_{j}$ are the normalized inter-frame predicted and ground-truth bounding box targets (Feichtenhofer et al., 2017). We only use N_{tra} ground-truths (track rois) from first frame $t - \delta$, that also have corresponding boxes (same Id) in frame t. At inference time, we use all detections (N) from the first frame as track rois. In the identity loss, q_{j,i^*} is the predicted probability of roi j for its ground-truth ID i^*. This loss is only computed for the foreground rois that also have an associated ground-truth identity i^* (N_{iden}). The loss weights λ_x are set to 1.

3.2. Online Tracking

We implement a simple tracking-by-detection algorithm that associates the detection hypotheses to target trajectories. Our algorithm creates a tracking buffer[1]. In order to assign each detection at frame t to a correct trajectory at time $t-1$, we build a bipartite graph between all detections D^{t} and all trajectories T^{t-1}, followed by Hungarian algorithm to find the optimal set of one-to-one assignment. The association weights of the graph edges are given by an affinity matrix.

In this work, we analyze two types of association metrics: (i) bounding box intersection-over-union (IoU) based association, and (ii) ID-feature embedding based association, where cosine similarity is utilized to compute the association. In comparison to ID-features, utilizing IoU metric enables complementary

[1] Tracking buffer T^{t-j} contains the trajectory information on all the trajectories at time $t - j$. Trajectory information includes (a) trajectory head bounding box, (b) trajectory head appearance feature vector, (c) average velocity of the target

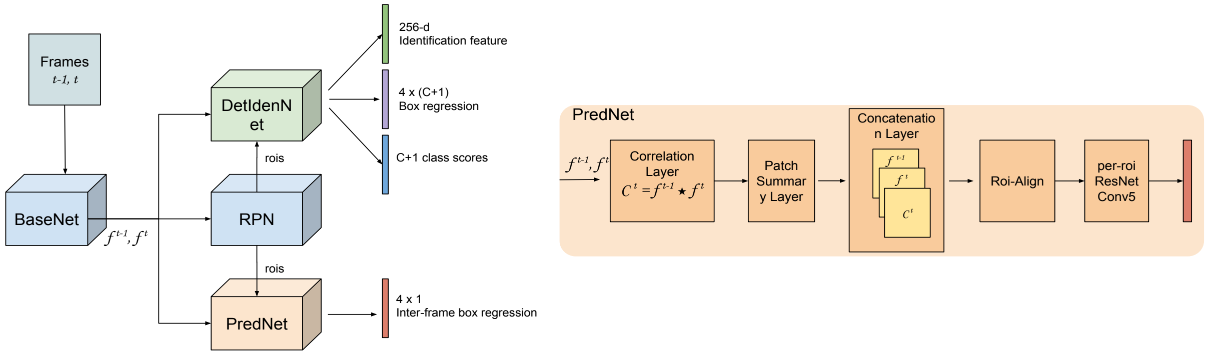


Fig. 2. Our model takes two frames as input and gives the detections with corresponding ID features and predictions via two parallel branches *DetIdenNet* and *PredNet*, respectively. *PredNet* computes the neighbourhood correlation of two base features at each feature location, followed by a patch summary layer. The correlated feature map and original base features are then used together to predict the inter-frame regression targets from first frame detections to the second.

spatial prior for the tracking step. Thus it is avoided to match wrong objects, that have similar appearance, for instance, cars of the same model/brand, or people wearing similar outfits.

In our evaluations we find that a simple combination of both of these metrics perform better than either individually. The overall association metric (A^t) is a weighted combination of these two metrics as given in Eq. 3. The weights w_1 and w_2 are set to 0.5 in our case. More details on the association metrics are given in the ablation study (Section 4.4).

$$A^t = w_1 * A^t_{IoU} + w_2 * A^t_{ID} \tag{3}$$

3.2.1. *Occlusion handling.*

To make our tracker robust to occlusions and missing detections/predictions, we use the buffer-and-recover approach to pause those trajectories that were not associated to any new detection. Such paused trajectories are maintained in a buffer (of size $buffer\_size$) and are available for matching later. For the detections at frame t that did not get associated to any trajectory from frame $t-1$, we compare their ID features to the unassigned trajectories from frame $t-2$, $t-3$ and so on. In this way, the objects that reappear after occlusion can be associated to the correct trajectory in past. All unassigned detections in frame t are then assigned new Ids and initialize new trajectories.

To reduce the fragmentations in trajectories, we also use a simple linear motion model to propagate the paused trajectories for a short period (*e.g.* 5 frames) of time in addition to the per-frame predictions from the $PredNet$.

4. Experiments

In this section, we provide detailed experimental analysis of our framework and its components. We also show the performance of the proposed architecture on the challenging UA-DETRAC (Wen et al., 2015; Lyu et al., 2017) and MOT (Milan et al., 2016) benchmark databases.

4.1. *Dataset*

UA-DETRAC challenge dataset contains 100 videos of objects belonging to 4 categories (car, bus, van, others). The training set contains 60 videos, while rest of the 40 videos are used for testing. The videos are captured in different traffic scenarios and weather conditions (cloudy, sunny, night and rainy) at 25 fps with 960 x 540 image resolution. For validation purpose, we take out a subset of 20 videos from the training set. As in (Amin and Galasso, 2017), we select ground-truths with occlusion and truncation ratio ≤ 0.5. We further filter the track IDs (for OIM) that are present in at least three frames giving us total of 55,093 training images and 3,949 identities.

4.2. *Training and Testing Details*

Our complete model is built upon $ResNet$-101 ($d = 1024$) and is trained similarly to (Amin and Galasso, 2017) to handle the numerous small objects in UA-DETRAC dataset. More specifically, we use a reduced-stride Conv4 block, providing finer resolution features, i.e. 1/8 height and width of the original image and smaller receptive field. We also use 15 anchors at each position of RPN, corresponding to 5 scales $\{2, 4, 8, 16, 32\}$ and 3 aspect ratios $\{0.5, 1, 2\}$ as opposed to Vanilla Faster RCNN (Ren et al., 2015) that uses 9 anchors. We scale the input images so that the shorter side is 600 pixels as long as the larger side is less than 1000 pixels (Girshick, 2015). For the Correlation Layer, we use window size of 5×5 ($n = 2$). At the time of training, we pass frames I^{t-1} and I^t of a video sequence ($\delta = 1$). We train using Stochastic Gradient Descent with an initial learning rate of 10^{-3} for 5 epochs followed by learning rate of 10^{-4} for 2 epochs. We train the model together with the Region Proposal Network (RPN) using joint optimization scheme rather than alternative optimization (Ren et al., 2015). At the time of testing, we do NMS with IoU threshold of 0.3. Our model, implemented in Pytorch, runs on an NVIDIA Quadro P6000 GPU at 3fps.

4.3. *Detection Evaluation*

We aim to build our tracking approach upon a strong detection framework. First we evaluate the detection performance of our model on our selected DETRAC *val* set.

Figure 3 illustrates the precision-recall of the detection results for different combinations of the multi-task objective. It may be noted from the plot the relative lower performance of

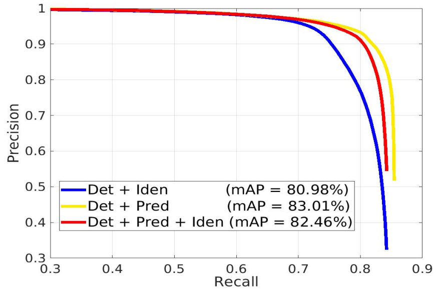


Fig. 3. Precision-recall curves and mean average precision (mAP) comparing detection results, as evaluated on the selected UA-DETRAC validation set. The joint detection+tracking+identification model (red curve) stands close to the detection+tracking (yellow) curve, quite above the detection+identification (blue) curve.

detection, when trained jointly with identification (*blue* curve), as compared to the detection performance, when trained jointly with tracking (*yellow* curve). Our proposed joint model (detection, tracking and identification jointly optimized, *red* curve) yields detection results close to the better detection and tracking results. In terms of average precision, our proposed joint model is only 0.5% from the detection+tracking setup, but 1.5% better than the detection+identification setup. While providing state-of-the-art detection performance, the joint model comes with the clear advantage of a single network, both for training and inference, end-to-end trained.

4.4. *Tracking Evaluation*

In order to quantify the effect of each component of the proposed architecture (see Section 3.1) on the tracking performance, we experimented on the DETRAC *val* set using the model that is learned on the DETRAC *train* set. As we have 3 main components in our architecture, we started with the simplest model that can be used for tracking purposes, i.e. detection with IoU association (“Asso.”) , and added another component one at a time to the evaluation process, so that we could see how the results are affected with each component. The 3 models we examined:

Model 1 (Detection + IoU Asso.): To see the effect of prediction given by our model, we first ignored the prediction and only used the detections for IoU asscociation (Section 3.2) to calculate different tracking evaluation metrics.

Model 2 (Detection + Prediction + IoU Asso.) : Next, we added the predictions given by the tracking branch of our model for the IoU association: for each detection at a frame, the corresponding bounding box at the successive frame is regressed by the tracking branch and used for the IoU computation.

Model 3 (Detection + Prediction + Identification Feature + ID Asso.): Finally, we added identification features in the IoU estimation. Here we regress the bounding boxes in the next frame thanks to the tracking branch, as in model 2, but we compute associations using both IoU’s and identification features (cf. Sec. 3.2).

We report the results of these 3 main components for both low and high (full) frame rates. Typical trackers assume sufficiently high detection frame rate, however this assumption is often violated in real-world settings with low cost and power constraints. On a standard low cost embedded GPU device such as NVIDIA Jetson TX2 (jetson tx2) state-of-the-art object detectors like Faster R-CNN (Ren et al., 2015) achieve only 1 fps. We challenge our proposed model for such low frame rate settings and substantially lowered the original frame rate 10 times, resulting to only 2.5 fps.

Table 1. Ablation results for 25fps and 2.5fps on DETRAC *val* set. For each method, the maximum values of the metrics across 9 detection thresholds (cf. Figure 4) are provided. ↑ means the higher the value, the better the performance, whereas ↓ means the lower the value, the better the performance. For each column, the best values are shown in bold.

Model	MOTA ↑	MOTP ↑	IDS ↓	MT ↑	ML ↓	Frag ↓	FP ↓	FN ↓	fps
Det + *IoU Asso.*	69.47	**85.85**	836	1259	167	2316	**12169**	47218	25
Det + Pred + *IoU Asso.*	69.49	85.83	597	1265	159	2164	12863	46718	25
Det + Pred+ Iden + *ID Asso.*	**69.50**	85.82	**592**	**1265**	**159**	**2140**	12946	**46636**	25
Det + *IoU Asso.*	57.17	**85.55**	2298	1098	160	384	**1127**	5327	2.5
Det + Pred + *IoU Asso.*	55.14	85.54	2311	1100	160	384	1539	5318	2.5
Det + Pred+ Iden + *ID Asso.*	**65.29**	85.48	**164**	**1120**	**150**	**416**	1830	**5099**	2.5

Table 1 shows the performance gain of adding each component for 25 and 2.5 fps, respectively. As the detection threshold can play a major factor in performance, we tested each component for 9 different detection thresholds of {0.1, ..., 0.9}, and report maximum values for 8 different metrics, namely Multiple Object Tracking Accuracy (MOTA), Multiple Object Tracking Precision (MOTP), Id Switches (IDS), Mostly Tracked (MT), Mostly Lost (ML), Fragmentations (Frag), False Positives (FP) and False Negatives (FN).

We observe that for 25 fps the MOTA and MOTP values reported in Table 1 are almost the same for all tested models. One of the reasons for this is, for instance, the dominating factor in MOTA, i.e. the number of FNs, which is heavily affected by the performance of the detector (Leal-Taixé et al., 2017). And since we used the same detector in each model, in order to better understand the contribution of each component, MOTA and MOTP values are expected not to change much. However there is an interesting improvement in the MOTA for the case of Model 3 for low frame rate. We observe that IDS decreased from 2298 to 164 which lead to significant improvement in MOTA, by around 8 points. For a more comprehensive evaluation, in Figure 4, we show MOTA vs. detection threshold plots for 25 and 2.5 fps, respectively. For the low fame rate, it is observed that the proposed model (Model 3) is robust to the changes in detection threshold and provides the best MOTA value for any given threshold. For high frame rate, we observe that Model 2 is doing as good as Model 3, which is expected as the detection bounding boxes from consecutive frames for a high frame rate overlap greatly, i.e. $IoU > 0.5$.

In Table 1, Model 3 results as the clear best performer across all metrics which are concerned with tracking only, namely IDS, MT, MK, Frag. Furthermore, we show that adding predictions to detections in the case of IoU Asso. (Model 2 vs. Model 1) improves the tracker performance in all tracking metrics.

4.5. *Comparative Study*

We compared our final model (Model 3) with the state-of-the art detection-and-tracking work by Feichtenhofer in (Feichtenhofer et al., 2017). This architecture jointly learns detection and prediction tasks (similarly to our Model 2, cf. Section 4.4). We

Table 2. Comparison of the proposed work (Model 3) with the current state-of-the-art method in (Feichtenhofer et al., 2017) on DETRAC *val* set for 25 fps and 2.5 fps. For each method, the maximum values of the metrics across 9 detection thresholds (cf. Figure 4) are provided. ↑ means the higher the value, the better the performance, whereas ↓ means the lower the value, the better the performance. For each column, the best values are shown in bold.

Model	MOTA ↑	MOTP ↑	IDS ↓	MT ↑	ML ↓	Frag ↓	FP ↓	FN ↓	fps
(Feichtenhofer et al., 2017)+*IoU Asso.*	**70.63**	**87.60**	**514**	1206	183	**1360**	**7761**	49652	25
Model 3	69.50	85.82	592	**1265**	**159**	2140	12946	**46636**	25
(Feichtenhofer et al., 2017)+*IoU Asso.*	56.56	**87.03**	2264	1054	185	**232**	**1136**	5476	2.5
Model 3	**65.29**	85.48	**164**	**1120**	**150**	416	1830	**5099**	2.5

trained their model on the same DETRAC *train* set and tested against ours on the DETRAC *val* dataset.

In contrast to the ablation study conducted above, we evaluate jointly the detection and tracking steps, as each model has its own trained detector in Table 2.

For high frame rate case, there is a slight difference in MOTA of our proposed method (Model 3) in comparison to (Feichtenhofer et al., 2017). We believe, this is due to a slight gap in the underlying detector performance (cf. Fig. 3). But for the 2.5 fps, the proposed model improves MOTA by around 9 points compared to the competitor. Both models report decrease in performance (in MOTA) once the database changes from 25 fps to 2.5 fps, which is understandable as the task becomes much harder.

Furthermore, Figure 4 compares the performance of the two architectures in terms of MOTA vs. detection threshold values for 25 and 2.5 fps, respectively. As expected, for high frame rate, there is no clear winner. However, for the case of 2.5 fps, the proposed model outperforms the model in (Feichtenhofer et al., 2017). This gap in performance is much more evident for the case of 2.5 fps, which shows that, given a low frame rate video, ID features are more reliable than the predicted bounding box locations.

4.6. *Challenge Participations*

4.6.1. *DETRAC Challenge*

Finally we trained our complete model on all 60 videos of UA-DETRAC dataset and submitted our results to 2018 AVSS challenge (see Table 3). Our online tracker ranked third in the challenge for private detections. Furthermore to our knowledge, our method ranked first among the online trackers.

Our proposed method achieved better performance than the 2017 challenge winner EB+IOUT (Bochinski et al., 2017), which is an offline tracker (combined with detections by the Evolving Boxes (EB) (Wang et al., 2017)).

Table 3. Tracking performance of the proposed model on UA-DETRAC (AVSS) 2018 challenge. Our method is shown in bold. Symbol (-) indicates that it is not clear if the corresponding method is an online or offline one.

Model	PR-MOTA ↑	PR-MOTP ↑	PR-MT ↑	PR-ML ↓	PR-IDS ↓	PR-Frag ↓	PR-FP ↓	PR-FN ↓
GP-FRCNN (Amin and Galasso, 2017) + Ours (Online)	**28.0**	**41.8**	**34.2**	20.9	698	3432	55801	150493
RCNN (Girshick et al., 2014) + MFOMOT(Online)	14.8	35.6	11.9	**20.8**	870	**2035**	**21277**	151788
CompACT (Cai et al., 2015) + GMMA (Online)	12.3	34.3	10.8	21.0	**628**	2424	25577	**144149**
RD^2 (Zhang et al., 2018) + KF-IOU (-)	**31.0**	**49.9**	**37.4**	**10.4**	725	996	52243	**94728**
FRCNN (Ren et al., 2015) +V-IOU (Offline)	29.0	35.8	30.1	22.2	**142**	**244**	14177	143880
EB+IOUT(Bochinski et al., 2017) (Offline)	16.4	26.7	14.8	18.2	1743	1846	**12627**	136078
CompACT (Cai et al., 2015) + DMC (-)	14.6	34.1	11.6	20.6	908	1287	16057	141463

Table 4. Tracking performance of the proposed model on MOT16 benchmark in comparison to the best online (causal) trackers.

Detection	Method	MOTA ↑	IDF1 ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓	Frag ↓
Public	BnW *(Not Published)*	**53.6**	52.8	**19.0**	36.6	5217	**78471**	909	1742
	Ours	49.7	46.8	16.7	37.3	4393	86241	1040	3652
	MOTDT (Long et al., 2018)	47.6	50.9	15.2	38.3	9253	85431	792	1858
	JCSTD (Tian et al., 2019)	47.4	41.1	14.4	**36.4**	8076	86638	1266	2697
	AMIR (Sadeghian et al., 2017)	47.2	46.3	14	41.6	**2681**	92856	774	1675
	DMMOT (Zhu et al., 2018a)	46.1	**54.8**	17.4	42.7	7909	89874	532	1616
	STAM16 (Chu et al., 2017)	46.0	50.0	14.6	43.6	6895	91117	**473**	1422
	RAR16pub(Fang et al., 2017)	45.9	48.8	13.2	41.9	6871	91173	648	1992
	DCCRF16(Zhou et al., 2018a)	44.8	39.7	14.1	42.3	5613	94133	968	**1378**
	TBSS(Zhou et al., 2018b)	44.6	42.6	12.3	43.9	4136	96128	790	1419
Private	PAAS *(Not Published)*	**69.6**	68.6	38.6	**17.9**	9138	**45497**	768	1969
	POI (Yu et al.)	66.1	65.1	34.0	20.8	5061	55914	805	3093
	CNNMTT (Mahmoudi et al., 2018)	65.2	62.2	32.4	21.3	6578	55896	946	2283
	TAP (Zhou et al., 2018c)	64.8	**73.5**	**40.6**	22.0	13470	49927	794	**1050**
	RAR16wVGG (Fang et al., 2017)	63.0	63.8	39.9	22.1	13663	53248	**482**	1251
	DeepSORT (Wojke et al., 2017)	61.4	62.2	32.8	18.2	12852	56668	781	2008
	SORT (Bewley et al., 2016)	59.8	53.8	25.4	22.7	8698	63245	1423	1835
	Ours	55.3	50.7	24.5	26.0	12309	68312	1320	3609
	EAMTT (Sanchez-Matilla et al., 2016)	52.5	53.3	19.0	34.9	**4407**	81223	910	1321

The first and second ranked methods, i.e. RD^2+KF-IOU and FRCNN+V-IOU, in AVSS challenge are shown in Table 3. FRCNN+V-IOU is an offline tracker that uses IOU criteria on the object bounding boxes from Faster R-CNN detector along side with visual features from these bounding boxes to do tracking. Though, this is an offline tracker, unlike ours. That is, to achieve such good performance, it post-processes all the trajectories to ensure that all final tracks start and end with a good detection bounding box. RD^2+KF-IOU is a tracker that combines RD^2-based detections along with simple Kalman filter. Their good tracking accuracy seems mostly due to their good detections. Compared to the detector we used in this paper (GP-FRCNN), RD^2 achieves 8.78% better detection performance in the AVSS detection challenge and it ranks first.

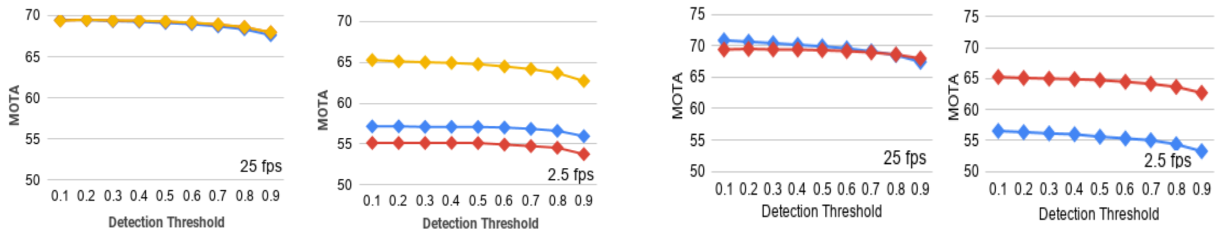
4.6.2. *MOT16 Benchmark*

In order to show the generalizability of the proposed model, we also tested our joint framework on the pedestrian tracking problem. For this experiment, we empirically set the weight of ID features in Equation 3 much higher than the one of IoU, i.e. $w_2 = 0.8$ and $w_1 = 0.2$. The reason for this choice was that the MOT16 database (Milan et al., 2016), we use in this experiment, was very challenging with lots of partial and full occlusions on pedestrians.

Table 4 shows the tracking results of our final model in comparison to other online trackers using both public and private detections. Among published online trackers using public detections, we rank first with 49.7% MOTA, while there is an unpublished entry with 53.6% MOTA. Among online trackers using private detections, we achieve a limited MOTA of 55.3%. It is critical to state that other trackers employ a separate detector to obtain the detections, whereas our model uses detections from the joint framework. Though, there is still room for improvement both in detection and tracking aspects.

5. Conclusion and future work

We have proposed a detection and tracking approach based on the joint end-to-end optimization of detection, tracking and identification. We have shown that the three tasks make compatible multi-task objectives, when adapted to videos. The simple integration of detections and tracking associations into an IoU-based tracker results in the best or comparable performance to other leading online trackers in the UA-DETRAC and MOT challenges.



(a)

(b)

Fig. 4. (a) shows the MOTA curves for the models in Table 1. Model 1, 2 and 3 are shown in blue, red and yellow, respectively. In the plot (a) left the three curves nearly overlap behind the yellow one. (b) shows MOTA curves for the models in Table 2. Our model 3 and method by Feichtenhofer et al. with IOU Asso. are shown in red and blue, respectively.

Acknowledgments

This research was partially supported by BMWi - Federal Ministry for Economic Affairs and Energy (MEC-View Project).

References

Almazán, J., Gajic, B., Murray, N., Larlus, D., 2018. Re-id done right: towards good practices for person re-identification. CoRR abs/1801.05339.

Amin, S., Galasso, F., 2017. Geometric proposals for faster r-cnn, in: AVSS, IEEE. pp. 1–6.

Benenson, R., Omran, M., Hosang, J., Schiele, B., 2014. Ten years of pedestrian detection, what have we learned?, in: ECCV, pp. 613–627.

Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B., 2016. Simple online and realtime tracking, in: ICIP, IEEE. pp. 3464–3468.

Bochinski, E., Eiselein, V., Sikora, T., 2017. High-speed tracking-by-detection without using image information, in: AVSS, 2017, IEEE. pp. 1–6.

Cai, Z., Saberian, M.J., Vasconcelos, N., 2015. Learning complexity-aware cascades for deep pedestrian detection. CoRR abs/1507.05348.

Chen, W., Chen, X., Zhang, J., Huang, K., 2017. Beyond triplet loss: a deep quadruplet network for person re-identification, in: CVPR.

Chu, Q., Ouyang, W., Li, H., Wang, X., Liu, B., Yu, N., 2017. Online multi-object tracking using cnn-based single object tracker with spatial-temporal attention mechanism. 2017 ICCV , 4846–4855.

Fang, K., Xiang, Y., Savarese, S., 2017. Recurrent autoregressive networks for online multi-object tracking. CoRR abs/1711.02741.

Feichtenhofer, C., Pinz, A., Zisserman, A., 2017. Detect to track and track to detect, in: ICCV.

Girshick, R., 2015. Fast r-cnn, in: ICCV.

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation, in: CVPR.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: CVPR, pp. 770–778.

Jifeng Dai, Yi Li, K.H., Sun, J., 2016. R-fcn: Object detection via region-based fully convolutional networks, in: NIPS.

Joseph Redmon, Santosh Divvala, R.G.A.F., 2016. You only look once: Unified, real-time object detection, in: CVPR.

Kaiming He, Georgia Gkioxari, P.D., Girshick, R., 2017. Mask r-cnn, in: ICCV.

Kalayeh, M.M., Basaran, E., Gokmen, M., Kamasak, M.E., Shah, M., 2018. Human semantic parsing for person re-identification, in: CVPR.

Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., . . . , Zhang, Z., He, Z., 2017. The visual object tracking vot2017 challenge results, in: ICCVW, pp. 1949–1972.

Leal-Taixé, L., Milan, A., Schindler, K., Cremers, D., Reid, I.D., Roth, S., 2017. Tracking the trackers: An analysis of the state of the art in multiple object tracking. CoRR abs/1704.02781.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C., 2016. Ssd: Single shot multibox detector, in: ECCV, Springer. pp. 21–37.

Long, C., Haizhou, A., Zijie, Z., Chong, S., 2018. Real-time multiple people tracking with deeply learned candidate selection and person re-identification, in: ICME.

Lyu, S., Chang, M., Du, D., Wen, L., Qi, H., . . . , , Huang, Y., Zhang, Y., 2017. Ua-detrac 2017: Report of avss2017 iwt4s challenge on advanced traffic monitoring, in: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 1–7.

Mahmoudi, N., Ahadi, S.M., Rahmati, M., 2018. Multi-target tracking using cnn-based features: Cnnmtt. ultimedia Tools and Applications 1-20.

Milan, A., Leal-Taixé, L., Reid, I., Roth, S., Schindler, K., 2016. Mot16: A benchmark for multi-object tracking. arXiv preprint arXiv:1603.00831 .

Real, E., Shlens, J., Mazzocchi, S., Pan, X., Vanhoucke, V., 2017. Youtube-boundingboxes: A large high-precision human-annotated data set for object detection in video, in: CVPR, pp. 7464–7473.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. NIPS , 91–99.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al., 2015. Imagenet large scale visual recognition challenge. IJCV 115, 211–252.

Sadeghian, A., Alahi, A., Savarese, S., 2017. Tracking the untrackable: Learning to track multiple cues with long-term dependencies, in: ICCV.

Sanchez-Matilla, R., Poiesi, F., Cavallaro, A., 2016. Online multi-target tracking with strong and weak detections, in: Computer Vision - ECCV 2016 Workshops - Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part II, pp. 84–99.

Tian, W., Lauer, M., Chen, L., 2019. Online multi-object tracking using joint domain information in traffic scenarios. IEEE Transactions on Intelligent Transportation Systems PP, 1–11. doi:10.1109/TITS.2019.2892413.

jetson tx2, . jetson-tx2. %urlhttps://developer.nvidia.com/embedded/buy/jetson-tx2.

Wang, L., Lu, Y., Wang, H., Zheng, Y., Ye, H., Xue, X., 2017. Evolving boxes for fast vehicle detection, in: ICME, pp. 1135–1140.

Wen, L., Du, D., Cai, Z., Lei, Z., Chang, M.C., Qi, H., Lim, J., Yang, M.H., Lyu, S., 2015. Ua-detrac: A new benchmark and protocol for multi-object detection and tracking. arXiv preprint arXiv:1511.04136 .

Wojke, N., Bewley, A., Paulus, D., 2017. Simple online and realtime tracking with a deep association metric. arXiv preprint arXiv:1703.07402 .

Wu, Y., Lim, J., Yang, M.H., 2015. Object tracking benchmark. TPAMI 37.

Xiao, T., Li, S., Wang, B., Lin, L., Wang, X., 2017. Joint detection and identification feature learning for person search, in: CVPR, IEEE. pp. 3376–3385.

Yu, F., Li, W., Li, Q., Liu, Y., Shi, X., Yan, J., . Poi: multiple object tracking with high performance detection and appearance feature, in: ECCV.

Zhang, S., Wen, L., Bian, X., Lei, Z., Li, S.Z., 2018. Single-shot refinement neural network for object detection, in: CVPR.

Zhou, H., Ouyang, W., Cheng, J., Wang, X., Li, H., 2018a. Deep continuous conditional random fields with asymmetric inter-object constraints for online multi-object tracking. TCSVT , 1–1.

Zhou, X., Jiang, P., We, Z., Dong, H., Wang, F., 2018b. Online multi-object tracking with structural invariance constraint, in: BMVC.

Zhou, Z., Xing, J., Zhang, M., Hu, W., 2018c. Online multi-target tracking with tensor-based high-order graph matching. doi:10.1109/ICPR.2018.8545450.

Zhu, J., Yang, H., Liu, N., Kim, M., Zhang, W., Yang, M.H., 2018a. Online multi-object tracking with dual matching attention networks, in: ECCV.

Zhu, X., Dai, J., Zhu, X., Wei, Y., Yuan, L., 2018b. Towards high performance video object detection for mobiles. CoRR abs/1804.05830.