VAN: Versatile Affinity Network for End-to-end Online Multi-Object Tracking

Hyemin Lee[1][0000−0002−1899−7211], Inhan Kim[1][0000−0002−1426−108X], and Daijin Kim[1][0000−0002−8046−8521]

Department of Computer Science and Engineering, POSTECH, Pohang, Korea
{lhmin, kiminhan, dkim}@postech.ac.kr

Abstract. In recent years, tracking-by-detection has become the most popular multi-object tracking (MOT) method, and deep convolutional neural networks (CNNs)-based appearance features have been successfully applied to enhance the performance of candidate association. Several MOT methods adopt single-object tracking (SOT) and handcrafted rules to deal with incomplete detection, resulting in numerous false positives (FPs) and false negatives (FNs). However, a separately trained SOT network is not directly adaptable because domains can differ, and handcrafted rules contain a considerable number of hyperparameters, thus making it difficult to optimize the MOT method. To address this issue, we propose a versatile affinity network (VAN) that can perform the entire MOT process in a single network including target specific SOT to handle incomplete detection issues, affinity computation between target and candidates, and decision of tracking termination. We train the VAN in an end-to-end manner by using event-aware learning that is designed to reduce the potential error caused by FNs, FPs, and identity switching. The proposed VAN significantly reduces the number of hyperparameters and handcrafted rules required for the MOT framework and successfully improves the MOT performance. We implement the VAN using two baselines with different candidate refinement methods to demonstrate the effects of the proposed VAN. We also conduct extensive experiments including ablation studies on three public benchmark datasets: 2D MOT2015, MOT2016, and MOT2017. The results indicate that the proposed method successfully improves the object tracking performance compared with that of baseline methods, and outperforms recent state-of-the-art MOT methods in terms of several tracking metrics including MOT accuracy (MOTA), identity F1 score (IDF1), percentage of mostly tracked targets (MT), and FP.

Keywords: Multi-Object Tracking, Multiple-Target Tracking, Visual Object Tracking, Target Association, Similarity Learning

1 Introduction

Multi-object tracking (MOT) is a core problem in computer vision and appears in various fields, such as video surveillance, humancomputer interaction, and autonomous driving. In recent years, tracking-by-detection has become the most popular MOT method. This method associates external detection results with targets to construct object trajectories [1–6]. The main operation in tracking-by-detection is the computation of affinity scores between targets and detection candidates. Many researchers have attempted to utilize various types of information to compute the affinity, such as appearance features, motion features, and location features. Deep convolutional neural networks (CNNs) have been successfully applied to MOT methods to significantly improve the extraction of useful features for candidate associations to mitigate the identity-switching problem. CNNs for MOT are designed to identify targets in place of handcrafted similarity measures for computing the similarity between targets and candidates [7–10]. However, despite the benefits of CNNs, MOT methods still suffer from incomplete detection, leading to numerous false positives (FPs) and false negatives (FNs) that can even make it impossible to associate targets with candidates. One potential solution to this problem is to include supplementary candidates for incomplete detection. Several MOT methods generate additional candidates to complement missing candidates by predicting potential locations using motion models and independent single-object tracking (SOT) [11–13, 10, 14].

However, utilizing SOT for MOT prediction involves two problems. First, the integration of SOT is not straightforward because scores generated from SOT are not compatible with the affinity values between targets and candidates. Therefore, the integration of SOT requires additional handcrafted rules and hyperparameters that are difficult to be optimized. Additionally, a separately trained SOT does not guarantee optimal MOT accuracy because SOT is primarily designed to discriminate between targets and their surrounding backgrounds, leading to drift issues relative to other targets. Second, the additional candidate may cause another FPs if it is not carefully generated. SOT can worsen this FP problem if the target is initialized with the FP candidate and is not terminated immediately. In this case, SOT continuously tracks the FP, thereby increasing the FP until the tracking is terminated with a certain rule; this phenomenon is called ghost tracking.

To address these issues, we propose a versatile affinity network (VAN) that can perform the entire MOT process in a single network including target-specific SOT to handle incomplete detection, affinity computation between target candidates, and decision of tracking termination. This process is performed by making the SOT prediction scores compatible with the affinity values between targets and candidates. We trained the VAN in an end-to-end manner by using event-aware learning that is designed to reduce the potential error caused by FNs, FPs, and tracking termination. The proposed VAN significantly reduced the effort required to tune hyperparameters and did not require handcrafted rules for integrating the SOT and decision of tracking termination. The overall process of proposed method is illustrated in Fig. 1.

We performed extensive experiments to validate the effectiveness of the proposed method by using three public benchmark datasets: 2D MOT2015, MOT2016, and MOT2017 [15]. The proposed method outperformed existing state-of-the-art online MOT methods in terms of various metrics including MOT accuracy

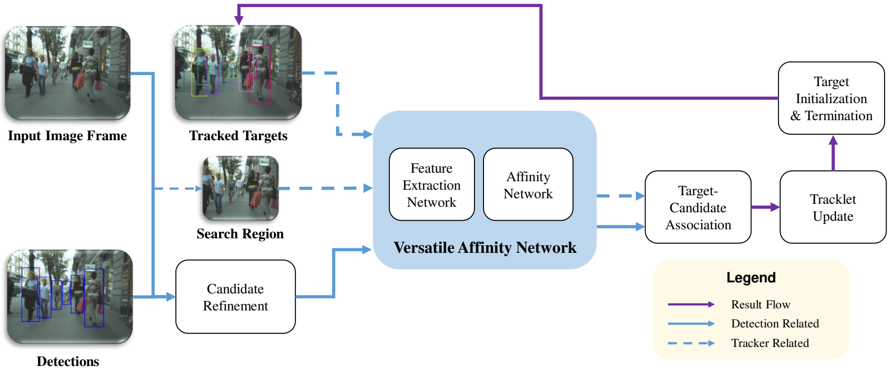


Fig. 1. Overview of a tracking method based on the proposed VAN.

(MOTA), identity F1 score (IDF1), percentage of mostly tracked targets (MT), and FNs. Additionally, we performed experiments to conduct ablation studies on a validation set by comparing a baseline method to variants of the proposed method to clarify how the proposed VAN improves tracking performance.

The main contributions of this study are as follows.

- We propose a novel VAN that is designed to perform the entire MOT process including SOT, affinity computation used to associate candidates, and decision of tracking termination.
- We train the VAN in an end-to-end manner by using event-aware learning that is designed to reduce the potential error caused by FNs, FPs, and tracking termination.
- We perform extensive experiments to demonstrate and verify that the proposed method improves the tracking performance.

2 Related Work

In recent years, the tracking-by-detection framework has become the most common MOT method. This framework solves an MOT problem as a detection association problem [1–6]. The core principle of tracking-by-detection is the association method and features used to compute the affinity between targets. Early MOT methods focused on the association method and used various optimization algorithms to solve the association problem.

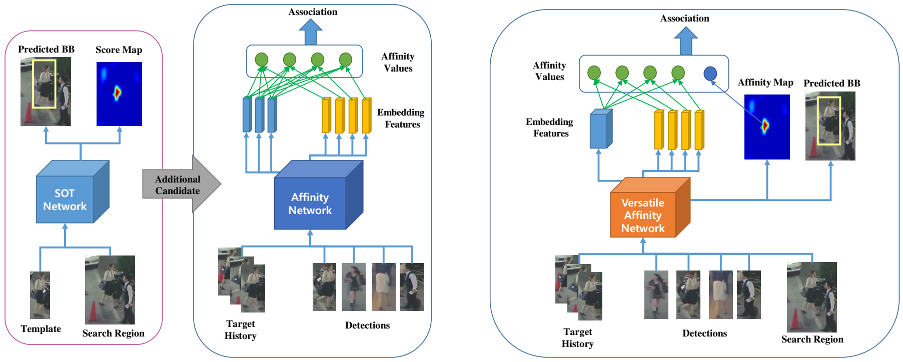
Offline trackers can utilize all frame information to determine a trajectory [16–18]. Therefore, they can use a global optimization method such as network flow [19–22], the Hungarian method [23, 24], and multiple hypothesis tracking [25, 26]. Offline trackers typically provide higher performance compared to online trackers. However, their application is limited because they cannot run in real-time. In contrast, online trackers can only utilize current and past frames. As an association is computed in every frame, linear assignment is frequently used to associate targets [7, 27, 28, 13]. Compared with offline trackers, online trackers tend to focus more on the features used to compute the affinity between targets.

In the recent studies on MOT, various features have been proposed to associate targets with candidates. These features can be categorized into three types, i.e., appearance, motion, and location. Affinity is defined using different combinations of these three features. [29] uses the location of candidates combined with neighbor, while other methods, such as [4, 9], combine appearance features with motion features.

The MOT methods developed in recent years have achieved success by utilizing deep CNNs, and they are designed to replace the conventional handcrafted affinity function to calculate the similarity between detection candidates. Certain methods [12, 8, 10] use deep features to associate candidates and long short-term memory (LSTM) [30, 29] to enhance discrimination features by utilizing temporal information. Recurrent neural networks (RNNs) are adopted to associate targets with candidates [9]. Owing to the use of large-scale datasets to re-identify the targets, CNN features can discriminate targets in various environments, such as occlusion, scale change, and reappearance. Recently, the authors of [31] adopt the classification and regression networks from an object detector and propose the regression-based tracking method that only utilizes object candidates as region proposals.

Even though deep CNNs successfully associate targets with candidates, the MOT problem remains difficult to solve owing to incomplete detections, which cause false alarms and false negatives. Even if an association is perfect, a missing target cannot be associated with any candidate if there are no suitable candidates. To address this problem, some MOT methods, such as [11, 10, 13, 18, 12], adopt SOT to overcome missing detection using SOT prediction as a complementary candidate. [11] uses a correlation-filter-based tracker [32] to achieve high speeds and utilizes SOT scores in a candidate decision process. CNN feature-based single object trackers, such as [33], are directly used in [10]. [12, 13] regard all detections as SOT candidates and develop an SOT module inside a framework. These approaches use a single object tracker to generate additional candidates, particularly for finding lost targets. However, SOT and MOT are not fully integrated; thus, the affinity network and single object trackers are generally trained independently.

Even though single object trackers can reduce false negatives, directly extending an SOT algorithm to an MOT problem is not straightforward because the former is mainly designed to discriminate a target from its surrounding background. In addition, as mentioned in Section 1, an SOT algorithm that exhibits high performance in an SOT task does not ensure high performance in an MOT task. Our method uses a VAN to compute the entire affinity scores for both SOT and MOT association task. This network naturally integrates SOT into an MOT association task by sharing target affinity, and as a result, the association becomes intuitive and tracking performance is improved.



a) previous method

b) proposed method

Fig. 2. Differences between methods using SOT: a) method with independent SOT and affinity networks, and b) proposed method with a versatile affinity network.

3 Proposed Method

3.1 Overview

In every frame, the proposed MOT method takes an image frame, detections, and target trajectories as inputs. First, detection candidates are refined to filter FP candidates by using two-class classifiers. The remaining candidates, previously tracked targets, and corresponding search regions are then passed through the VAN. In the feature extraction network, detection candidates and tracked targets with corresponding search regions are processed and CNN features are obtained. The extracted features can be reused in all tracking processes.

In an affinity network, target features, including initial appearance features and temporal features, are correlated with search regions and detection candidates to compute affinity scores. Based on the affinity score matrix, targets and candidates including SOT results are associated using an optimization algorithm, new target is initialized, and the target that disappears is terminated. Fig. 1 shows the overall process of the proposed method.

3.2 Candidate Refinement

False positive candidates are a critical problem in MOT because they generate continuous false trajectories when they are initialized as new targets, making association extremely difficult. These false positive candidates can be filtered by applying an additional classifier, as discussed in [34, 31]. The authors of [34] adopted the feature extraction portion of a region-based fully convolutional network (R-FCN) detector [35] with SqueezeNet [36] as a backbone network. Their classifier uses entire image frames for feature extraction and the detection candidates in a single frame share a feature map. The feature maps corresponding

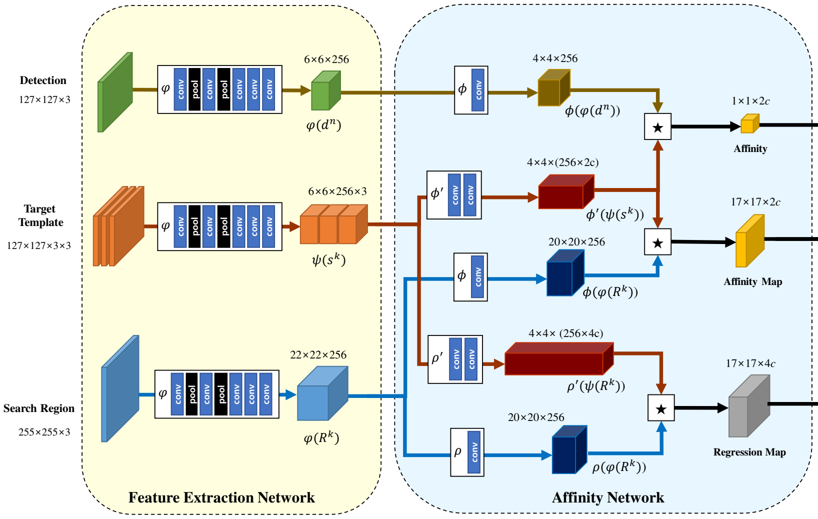


Fig. 3. VAN architecture. The feature extraction network shares layers and the template branch consists of different association layers for reflecting temporal features and anchors. The outputs of SOT branch are an affinity map and regression value map, whereas the output of the association branch is a single affinity value.

to each detection candidate are classified into two classes: background and object. The authors of [31] utilized the classification network of Faster R-CNN [37] with ResNet-101 [38] as a backbone network. Because the candidate refinement process can be viewed as part of the detection problem, we separate the performance gains from this process and implement our method using two algorithms as baseline trackers. Our method follows the same candidate refinement method and models used in [34, 31] to determine the pure contributions of our proposed VAN. Two baseline methods discussed above are denoted as RFCN and FRCNN respectively.

In our proposed method, the set of detection candidates in a t-th frame is denoted as $D_t = \{d_t\}$, where each detection is denoted as $d_t = \left\{d_t^x, d_t^y, d_t^w, d_t^h\right\}$. The features of an input image frame I_t are extracted using the backbone network. The classification scores corresponding to each candidate are calculated by applying ROI pooling with a softmax function in the final layer to filter candidates with scores lower than a threshold. Finally, non-maximum suppression (NMS) is applied based on the classification scores and remaining candidates are fed into the next step.

3.3 Versatile Affinity Network

In this subsection, we describe the proposed tracking framework based on the VAN and the overall network architecture. Tracking methods using SOT, such as those in [11–13, 18], typically derive predictions from previous frames utilizing SOT. SOT results are then considered as candidates and affinity calculation is performed by a different network (Fig. 2a). In this type of framework, the affinity computed by the SOT network cannot be directly used in the association step. Even if the calculated affinity is used, an additional cost function must be employed to make the scores obtained from SOT compatible with the affinity network. In the proposed method, all candidates share affinity values, regardless of whether they are detection candidates or candidates from SOT prediction. SOT prediction results can be directly adopted in the association and target management step. Therefore, additional hyperparameter tuning and heuristic rules are not required (see Fig. 2b).

The basic structure of the VAN follows that of the Siamese CNN proposed in [39]. In our versatile network, SOT uses classification and regression branches, whereas the affinity task only utilizes a classification branch. At t-th frame, the network takes an image frame I_t, a set pf previously tracked target trajectories from the previous frame $S^{(t-1)}$, and a set of detection candidates D_t as inputs. The image patches extracted from the target and detection candidate locations are resized to 127 × 127 pixels, and the search region for SOT is resized to 255 × 255 pixels. The feature extraction network consists of three convolutional layers and retains a fully convolutional structure. The feature extraction network of the Siamese CNN can take any size of input and shares all weights for inputs, including targets, detections, and search regions. Each tracked target s_t^k inside set $S^{(t)}$ maintains its corresponding features, $\varphi\left(s_t\right)$, through the tracking phase. Additionally, each target has its representative temporally concatenated features $\psi\left(s_t\right)$, which consist of features from initial target location $\varphi\left(s_1\right)$, intermediate location $\varphi\left(s_m\right)$, and very recent location $\varphi\left(s_t\right)$ where the intermediate location $m=[\frac{t+1}{2}]$. Initial features are fixed when a target is initialized, and intermediate features come features from the median frame index between the initial and most recent frames. Recent features are updated at every frame. Therefore, the k-th tracking target in frame t is denoted as $s_t=\left\{s_t^x, s_t^y, s_t^w, s_t^h, \varphi\left(s_t\right), \psi\left(s_t\right)\right\}$. This structure can enhance the robustness of target features, to handle occlusion and appearance changes. The feature extraction network is trained to generate embedding features suitable for comparing target features to candidate features. In other words, this network is optimized for computing correct affinity scores between targets and candidates. The size of the search region is twice that of each target's bounding box and the search region is denoted as R. To predict new locations using SOT from frames $t-1$ to t, the features of the search regions corresponding to each tracked target are extracted. Simultaneously, the features of detection candidates are also extracted. In summary, in the feature extraction phase, we extract feature set $\varphi\left(s^k\right)$, $\varphi\left(R^k\right)$ and $\varphi\left(d^n\right)$, where k and n are the indexes of the targets and detection candidates, respectively.

T

he representative features of a target and corresponding features from the search region pass through the affinity network to generate an affinity map and regression results, which are used to predict the next location of each target.

The features from a target s_k and its corresponding search region R_k are fed through the affinity layer and regression layer. We denote the output of the affinity layer as $\phi\left(\cdot\right)$ for the detection and search regions, and $\phi'\left(\cdot\right)$ for the temporally concatenated target template. The correlation operation is denoted as $\star$. The features $\phi'\left(\psi\left(s^k\right)\right)$ and $\phi\left(\varphi\left(R^k\right)\right)$ are used to calculate an affinity map A_k as follows:

$$A_k = \phi'\left(\psi\left(s^k\right)\right) \star \phi\left(\varphi\left(R^k\right)\right). \tag{1}$$

Among the affinity values, the best value corresponds to the predicted location for the next frame. This value is used in the association procedure. The regression value corresponding to the best location is applied to the current bounding box and the box is refined. The regression value indicates the normalized distance between the anchors and ground truth in the form of $\{dx, dy, dw, dh\}$. The regression map is calculated as follows:

$$E_k = \rho'\left(\psi\left(s^k\right)\right) \star \rho\left(\varphi\left(R^k\right)\right). \tag{2}$$

where the regression layers for the target template and the search region are denoted as $\rho'\left(\cdot\right)$ and $\rho\left(\cdot\right)$ respectively.

The detection features, $\varphi\left(d_t\right)$, pass through the affinity layer with target features and resulting in a $1 \times 1 \times 2c$ vector, where c is the number of anchors. The affinity between a single target s_k and detection d_n is calculated as

$$a_{nk} = \phi'\left(\psi\left(s^k\right)\right) \star \phi\left(\varphi\left(d^n\right)\right). \tag{3}$$

Because the purpose of the affinity network is to compare a target bounding box to a detection, we apply the softmax function across the channel to generate normalized affinity values. Among the resulting $1 \times 1 \times c$ affinity values, we only use the maximum value. The VAN computes the affinity values between all potential pairs of targets and detection candidates. In this manner, the network enables a tracker to perform both SOT and MOT associations. The network architecture and pipeline for this process were illustrated in Fig. 3.

3.4 Event-Aware Training

To train the proposed VAN, we utilized large-scale datasets containing ID information corresponding to each target to sample target and candidate pairs. The network is trained using the combination of classification loss and regression loss as proposed in [39]. We extracted target-candidate pairs from the YouTube-BB, ImageNet-VID, 2D MOT2015, and MOT2017 training sets. We randomly selected two frames from video sequences at 0 to 10 frame intervals and extracted positive samples with $IoU > 0.6$ and negative samples with $IoU < 0.4$ to make the network generalization power relative to the ground truth. Among selected two frames, target templates for temporally concatenated features are selected based on the ground truth with random perturbations.

Additionally, to make the network to have ability to discriminate the target with other candidate, we use event-aware training strategy motivated by

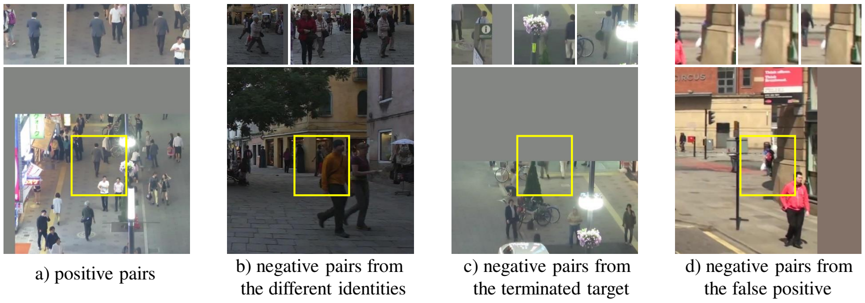

a) positive pairs
b) negative pairs from the different identities
c) negative pairs from the terminated target
d) negative pairs from the false positive

Fig. 4. Example of strategies to extract sample pairs for event-aware training.

distractor-aware training proposed in [40]. We designed the event-aware training suitable for MOT task to reduce the potential error caused by FNs, FPs, and tracking termination. To obtain semantically meaningful samples, we generated samples by using simulation trackers. The simulation trackers were pretrained using classical sampling technique. We ran the tracker with detection candidates, and calculated the affinity of whole target-candidate pairs. We decided the targets and candidates pairs included in a given event situation by using ground-truth assignments and extract sample pairs for preventing the situation. The event comprises three categories: false negative, false positive, and tracking termination. First, we extracted the negative pairs from two targets which has different identity (Fig. 4b). To extract the hard negative samples (distractors), we chose the candidate that had highest affinity except the true assignment. The FP could be reduced by degrading the SOT score if the SOT is initialized with FP candidate. We extracted the negative pairs from FP target generated during simulation (Fig. 4d). Also, the termination of track could arise another FP. Then, we explicitly cropped the negative pairs when the tracking was terminated by occlusion or exiting (Fig. 4c). These samples enabled the VAN to decide whether the tracking is terminated or not. Finally, the FN could be reduced by training the network elaborately using plenty of positive pairs within same identities (Fig. 4a).

3.5 Candidate Association

We associate targets and candidates by using the affinity values calculated in the previous section. We do not calculate affinity values for all possible pairs of targets and candidates because doing so would have been computationally expensive. We limit the possible change in targets (e.g., location and size). We calculate the affinity for pairs that satisfy this limitation and assign infinite negative affinity to other pairs. In this study, we apply different detailed tracking managements for each baseline methods.

For the RFCN baseline, we predict target locations using SOT and add the score to the affinity matrix. To prevent the SOT result being matched with other target, we assign infinite negative affinity to pairs that have different identity. Next, we compute the affinities for all potential target-candidate pairs, assign candidates only to the activated target using the Hungarian algorithm [24]. If the target is not associated with any candidate, and the SOT score is low, we deactivate this target. During this process, the SOT can naturally supplement missing detections without arising ghost tracking. Deactivated targets can associated with the remaining detection results for re-activation when these targets reappear on next frame.

For the FRCNN baseline, all targets are used to predict the next target based on SOT using the VAN. Targets are deactivated if the SOT score is not sufficient or the classification score is low. The affinity values for pairs of only deactivated targets and detection candidates are calculated using the VAN. If there exist additional matching, and the affinity is sufficiently high, a deactivated target is reactivated and is updated using the associated candidate location. The VAN is able to substitute the regression, classification, and reID module of the baseline tracker.

After all associations are completed, the remaining candidates that have no associations with any targets and exhibit low affinity value with all targets are added to the tracker and activated as new targets.

4 Experiments

We conducted several experiments to determine the effectiveness of the proposed VAN on three MOT benchmark datasets: 2D MOT2015, MOT2016, and MOT2017 [15]. The results of other trackers and the proposed method were evaluated using the official MOT challenge benchmark score board[1].

4.1 Implementation Details

The proposed method was implemented using PyTorch and tested on a workstation with a 6-core Intel i7@3.60 GHz CPU and NVIDIA Titan Xp GPU. We used an R-FCN architecture with SqueezeNet as the backbone network for the RFCN baseline and the Faster R-CNN detector with ResNet-101 as a backbone network for the FRCNN baseline. We followed the same tracking management strategy for both baselines, excluding the core tracking steps. The minimum threshold value for filtering candidates was set to 0.4. The VAN was implemented based on the Siamese CNNs [39] and the three convolutional layers of AlexNet [41] is utilized for feature extraction. The VAN was trained using stochastic gradient descent over 90 epochs with a learning rate of ranging from 10^{-2} to 10^{-6}. We generated training pairs from the YouTube-BB, ImageNet-VID, 2D MOT2015, and MOT2017 training sets with sets of two random frames extracted at intervals between 0 and 10 frames. The target template images were resized to 127×127 pixels and the search regions were resized to 255×255 pixels. We

[1] https://motchallenge.net

concatenated the initial features, recent features, and intermediate features of each targets to reflect the temporal attention of targets. In the initial frames, the intermediate and recent features were cloned from the initial features. For both baselines, the classification threshold for target initialization were set to 0.3 and the maximum lost time for termination is set to 30 frames. We limited the possible change of location as 1/10 of the diagonal length of the frame, and the possible size change as 1/3 of the previous target size.

4.2 Evaluation on MOT Benchmarks

The proposed method was evaluated on the 2D MOT2015, MOT2016, and MOT2017 test datasets using on an official website. We adopted the CLEAR MOT metrics [58] to evaluate the performance of the tracker on the MOT datasets and compare it with other state-of-the-art trackers. The representative metric was multiple object tracking accuracy (MOTA), which reflects the false negatives (FN), false positives (FP), and identity switches (IDS). Other metrics are also reported, including identity F1 scores (IDF1), percentage of mostly tracked targets (MT), mostly lost targets (ML). We implement two version of trackers corresponding to the baseline approach which are denoted as VAN(RFCN) and VAN(FRCNN) respectively.

The 2D MOT2015 test dataset consists of 11 video sequences obtained from various scene with ACF detection results. The tracking performance evaluated on 2D MOT2015 test dataset are listed in Table 1. Note that in case of [34], we evaluated the results ourselves because there are no official results on 2D MOT2015 dataset. The MOT2016 test dataset contains 7 videos that are entirely disjoint with the training set with DPM detection results. The results obtained for the MOT2016 test dataset are reported in Table 2. The MOT2017 test dataset contains the same video sequences as the MOT2016 dataset, but different detections are provided. This dataset focuses on evaluating trackers based on various detection results. Three types of detectors are used in this dataset: DPM, SDP, and Faster-RCNN. The results for the MOT2017 test dataset are listed in Table 3. We evaluated the proposed tracker using the same network model and hyperparameters throughout the testing process.

Compared to the baseline methods, the proposed VAN exhibits significant improvements on every benchmark datasets. The proposed method also achieves excellent results in terms of MOTA, ML, and FN compared to existing state-of-the-art MOT methods, even offline methods that can utilize global optimization. In particular, our method significantly reduces the FN by integrating SOT prediction into the association step. The experimental results demonstrate the excellent performance of the proposed VAN.

We simply extended our method for comparisons with an offline method by using trajectory interpolation to complement the missing part of the trajectory by using neighbor frames. This method was denoted as VAN-off; it achieves higher performance than the online version.

Table 1. Tracking Performance on the 2D MOT2015 benchmark test set. Best in bold.

Type	**Method**	**MOTA↑**	**IDF1↑**	**MT↑**	**ML↓**	**FP↓**	**FN↓**	**IDS↓**
offline	JPDA_m [42]	23.8	33.8	5.0	58.1	6373	40084	**365**
	R1TA [43]	24.3	24.1	5.5	46.6	6644	38582	1271
	SCNN [8]	29.0	34.3	8.5	48.4	**5160**	37798	639
	MHT_DAM [26]	32.4	45.3	16.0	43.8	9064	32060	435
	JMC [44]	35.6	45.1	23.2	39.3	10580	28508	457
	VAN-off	**47.4**	**49.5**	**24.0**	**26.8**	6044	**25164**	1087
online	SCEA [28]	29.1	37.2	8.9	47.3	6060	36912	604
	MDP [13]	30.3	44.7	13.0	38.4	9717	32422	680
	AMIR [29]	37.6	46.0	15.8	26.8	7933	29397	1026
	AP [7]	38.5	47.1	8.7	37.4	**4006**	33203	586
	KCF [11]	38.9	44.5	16.6	31.6	7321	29501	720
	Base (RFCN) [34]	33.1	44.3	9.1	46.2	6806	36226	615
	VAN (RFCN)	34.7	45.9	10.5	47.8	6907	32698	**540**
	Base (FRCNN) [31]	44.1	46.7	18.0	**26.2**	6477	**26577**	1318
	VAN (FRCNN)	**46.0**	**48.3**	**19.3**	28.4	4531	27340	1280

Table 2. Tracking Performance on the MOT2016 benchmark test set. Best in bold.

Type	**Method**	**MOTA↑**	**IDF1↑**	**MT↑**	**ML↓**	**FP↓**	**FN↓**	**IDS↓**
offline	MHT_DAM [26]	45.8	46.1	16.2	43.2	6412	91758	590
	NOMT [45]	46.4	53.3	**18.3**	41.4	9753	87565	**359**
	LMP [46]	48.8	51.3	18.2	40.1	6654	86245	481
	eTC [47]	49.2	56.1	17.3	40.3	8400	83702	606
	HCC [48]	49.3	50.7	17.8	39.9	5333	86795	391
	NOTA [49]	49.8	55.3	17.9	37.7	7248	83614	616
	VAN-off	**57.3**	**57.5**	**24.8**	**33.9**	**3845**	**73489**	550
online	oICF [50]	43.2	49.3	11.3	48.5	6651	96515	**381**
	STAM [12]	46.0	50.0	14.6	43.6	6895	91117	473
	DMAN [10]	46.1	**54.8**	17.4	42.6	7909	89874	532
	AMIR [29]	47.2	46.3	14.0	41.6	2681	92856	774
	KCF [11]	48.8	47.2	15.8	38.1	5875	86567	906
	Base (RFCN) [34]	47.6	50.9	15.2	38.3	9253	85431	792
	VAN (RFCN)	48.9	53.2	15.2	**36.2**	9987	82427	838
	Base (FRCNN) [31]	54.4	52.5	19.0	36.9	3280	**79149**	682
	VAN (FRCNN)	**54.6**	54.2	**19.4**	**36.2**	**2307**	79895	619

Table 3. Tracking Performance on the MOT2017 benchmark test set. Best in bold.

Type	Method	MOTA↑	IDF1↑	MT↑	ML↓	FP↓	FN↓	IDS↓
offline	IoU17 [51]	45.5	39.4	15.7	40.5	19993	281643	5988
	EDMT [52]	50.0	51.3	21.6	36.3	32279	247297	2264
	TLMHT [53]	50.6	56.5	17.6	43.4	22213	255030	**1407**
	MHT_DAM [26]	50.7	47.2	20.8	36.9	22875	252889	2314
	JCC [54]	51.2	54.5	20.9	37.0	25937	247822	1802
	FWT [55]	51.3	47.6	21.4	35.2	24101	247921	2648
	VAN-off	**57.4**	**57.9**	**26.3**	**33.7**	**14316**	**224064**	1788
online	PHD_GSDL [56]	48.0	49.6	17.1	35.6	23199	265954	3988
	AM_ADM [57]	48.1	52.1	13.4	39.7	25061	265495	2214
	DMAN [10]	48.2	**55.7**	19.3	38.3	26218	263608	2194
	HAM_SADF [57]	48.3	51.1	17.1	41.7	20967	269038	**1871**
	FAMNet [14]	52.0	48.7	19.1	33.4	14138	253616	5318
	Base(RFCN) [34]	50.9	52.7	17.5	35.7	24069	250768	2474
	VAN (RFCN)	52.0	53.9	**20.2**	**33.4**	31275	**237004**	2817
	Base(FRCNN) [31]	53.5	52.3	19.5	36.6	12201	248047	2072
	VAN (FRCNN)	**55.2**	54.2	20.0	35.5	**8522**	241848	2220

4.3 Ablation Study

We performed additional experiments to conduct ablation studies by using various versions of the proposed tracker to determine which modules affect tracking performance and to verify the effectiveness of the proposed approach. The experiments for the ablation studies were performed on a subset of the MOT2017 training dataset that was not used in training phase because the corresponding testing dataset did not provide ground truth labels for validation. We evaluated the SDP sequences of the MOT2017 dataset. We implemented five variants of each baseline tracker. The baseline tracker follows the existing MOT method without using SOT. The Base+SOT directly utilizes the SOT [39] and generates additional candidates for missing targets. Even when using SOT results without any fine tuning, the tracking performance of the RFCN baseline was improved. Note that the performance of Base+SOT for FRCNN baseline was degraded because this baseline uses a well-trained regression network for object detection, which has better performance than the raw SOT. Next, we trained the SOT module to improve the discrimination ability for the MOT datasets by training the networks using extra MOT datasets following the training approach of [39] while preserving the baseline association method. This approach is denoted as Base+TSOT in Table 4. This approach exhibits additional performance gains compared to the method directly using SOT. To demonstrate the effect of event-aware training and VAN architecture itself, we trained the proposed VAN without using event-aware learning strategy. Further, tracking termination were performed using existing methods. We denote this version as VAN-EA. This result shows the effectiveness of architecture of VAN itself to perform SOT and affinity computation. Finally, we utilized the proposed method. The VAN reduced the effort required to tune the hyperparameters and could significantly reduce FN and IDS with help of event-aware learning. These ablation studies prove that the proposed VAN and event-aware learning is a promising solution for MOT problems.

Table 4. Ablation study of various tracker versions on the MOT2017 benchmark validation set.

Base	**Method**	**MOTA↑**	**IDF1↑**	**MT↑**	**ML↓**	**FP↓**	**FN↓**	**IDS↓**
RFCN	Base	60.0	61.1	25.6	27.1	**1864**	42504	605
	Base+SOT	61.2	53.9	31.1	23.4	4092	38341	1099
	Base+TSOT	62.7	55.9	32.2	23.1	3503	37561	813
	VAN-EA	63.6	63.5	31.1	24.7	3057	37225	**539**
	VAN	**64.2**	**63.7**	**33.5**	**21.2**	3618	**35906**	685
FRCNN	Base	67.7	68.0	40.4	17.4	**803**	35055	368
	Base+SOT	63.6	62.9	42.3	16.7	4844	33414	2676
	Base+TSOT	66.6	66.5	39.0	17.3	1253	35704	520
	VAN-EA	68.6	69.2	43.5	17.2	1320	33578	322
	VAN	**69.1**	**70.7**	**44.3**	**16.6**	1671	**32656**	**305**

5 Conclusions

We proposed a novel MOT method using a VAN to perform the entire MOT process in a single network including SOT, affinity computation, and target management. During the tracking process, the results of target-specific SOT prediction and detection candidates are associated with targets by sharing network weights and compatible affinity values obtained from a unified network. The proposed method exhibited remarkable performance on several MOT benchmarks compared to state-of-the-art online MOT methods, making it a promising solution for MOT problems.

Acknowledgement

This work was supported by Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No.2014-0-00059, Development of Predictive Visual Intelligence Technology), (No.2017-0-00897, Development of Object Detection and Recognition for Intelligent Vehicles) and (No.2018-0-01290, Development of an Open Dataset and Cognitive Processing Technology for the Recognition of Features Derived From Unstructured Human Motions Used in Self-driving Cars)

References

1. Bae, S.H., Yoon, K.J.: Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2014) 1218–1225
2. Berclaz, J., Fleuret, F., Turetken, E., Fua, P.: Multiple object tracking using k-shortest paths optimization. IEEE transactions on pattern analysis and machine intelligence **33** (2011) 1806–1819
3. Brendel, W., Amer, M., Todorovic, S.: Multiobject tracking as maximum weight independent set. In: CVPR 2011, IEEE (2011) 1273–1280
4. Leal-Taixé, L., Fenzi, M., Kuznetsova, A., Rosenhahn, B., Savarese, S.: Learning an image-based motion context for multiple people tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2014) 3542–3549
5. Lenz, P., Geiger, A., Urtasun, R.: Followme: Efficient online min-cost flow tracking with bounded memory and computation. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 4364–4372
6. Milan, A., Schindler, K., Roth, S.: Multi-target tracking by discrete-continuous energy minimization. IEEE transactions on pattern analysis and machine intelligence **38** (2015) 2054–2068
7. Chen, L., Ai, H., Shang, C., Zhuang, Z., Bai, B.: Online multi-object tracking with convolutional neural networks. In: 2017 IEEE International Conference on Image Processing (ICIP), IEEE (2017) 645–649
8. Leal-Taixé, L., Canton-Ferrer, C., Schindler, K.: Learning by tracking: Siamese cnn for robust target association. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. (2016) 33–40
9. Milan, A., Rezatofighi, S.H., Dick, A., Reid, I., Schindler, K.: Online multi-target tracking using recurrent neural networks. In: Thirty-First AAAI Conference on Artificial Intelligence. (2017)
10. Zhu, J., Yang, H., Liu, N., Kim, M., Zhang, W., Yang, M.H.: Online multi-object tracking with dual matching attention networks. In: Proceedings of the European Conference on Computer Vision (ECCV). (2018) 366–382
11. Chu, P., Fan, H., Tan, C.C., Ling, H.: Online multi-object tracking with instance-aware tracker and dynamic model refreshment. In: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE (2019) 161–170
12. Chu, Q., Ouyang, W., Li, H., Wang, X., Liu, B., Yu, N.: Online multi-object tracking using cnn-based single object tracker with spatial-temporal attention mechanism. In: Proceedings of the IEEE International Conference on Computer Vision. (2017) 4836–4845
13. Xiang, Y., Alahi, A., Savarese, S.: Learning to track: Online multi-object tracking by decision making. In: Proceedings of the IEEE international conference on computer vision. (2015) 4705–4713
14. Chu, P., Ling, H.: Famnet: Joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking. In: Proceedings of the IEEE International Conference on Computer Vision. (2019) 6172–6181
15. Milan, A., Leal-Taixé, L., Reid, I., Roth, S., Schindler, K.: Mot16: A benchmark for multi-object tracking. arXiv preprint arXiv:1603.00831 (2016)
16. Kim, H.U., Kim, C.S.: Cdt: Cooperative detection and tracking for tracing multiple objects in video sequences. In: European Conference on Computer Vision, Springer (2016) 851–867

17. Pirsiavash, H., Ramanan, D., Fowlkes, C.C.: Globally-optimal greedy algorithms for tracking a variable number of objects. In: CVPR 2011, IEEE (2011) 1201–1208
18. Yan, X., Wu, X., Kakadiaris, I.A., Shah, S.K.: To track or to detect? an ensemble framework for optimal selection. In: European Conference on Computer Vision, Springer (2012) 594–607
19. Dehghan, A., Modiri Assari, S., Shah, M.: Gmmcp tracker: Globally optimal generalized maximum multi clique problem for multiple object tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 4091–4099
20. Leal-Taixé, L., Pons-Moll, G., Rosenhahn, B.: Everybody needs somebody: Modeling social and grouping behavior on a linear programming multiple people tracker. In: 2011 IEEE international conference on computer vision workshops (ICCV workshops), IEEE (2011) 120–127
21. Zamir, A.R., Dehghan, A., Shah, M.: Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs. In: European Conference on Computer Vision, Springer (2012) 343–356
22. Zhang, L., Li, Y., Nevatia, R.: Global data association for multi-object tracking using network flows. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition, IEEE (2008) 1–8
23. Huang, C., Wu, B., Nevatia, R.: Robust object tracking by hierarchical association of detection responses. In: European Conference on Computer Vision, Springer (2008) 788–801
24. Munkres, J.: Algorithms for the assignment and transportation problems. Journal of the society for industrial and applied mathematics **5** (1957) 32–38
25. Chu, P., Pang, Y., Cheng, E., Zhu, Y., Zheng, Y., Ling, H.: Structure-aware rank-1 tensor approximation for curvilinear structure tracking using learned hierarchical features. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer (2016) 413–421
26. Kim, C., Li, F., Ciptadi, A., Rehg, J.M.: Multiple hypothesis tracking revisited. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 4696–4704
27. Fagot-Bouquet, L., Audigier, R., Dhome, Y., Lerasle, F.: Online multi-person tracking based on global sparse collaborative representations. In: 2015 IEEE International Conference on Image Processing (ICIP), IEEE (2015) 2414–2418
28. Hong Yoon, J., Lee, C.R., Yang, M.H., Yoon, K.J.: Online multi-object tracking via structural constraint event aggregation. In: Proceedings of the IEEE Conference on computer vision and pattern recognition. (2016) 1392–1400
29. Sadeghian, A., Alahi, A., Savarese, S.: Tracking the untrackable: Learning to track multiple cues with long-term dependencies. In: Proceedings of the IEEE International Conference on Computer Vision. (2017) 300–311
30. Kim, C., Li, F., Rehg, J.M.: Multi-object tracking with neural gating using bilinear lstm. In: Proceedings of the European Conference on Computer Vision (ECCV). (2018) 200–215
31. Bergmann, P., Meinhardt, T., Leal-Taixe, L.: Tracking without bells and whistles. In: Proceedings of the IEEE International Conference on Computer Vision. (2019) 941–951
32. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. IEEE transactions on pattern analysis and machine intelligence **37** (2014) 583–596

33. Danelljan, M., Bhat, G., Shahbaz Khan, F., Felsberg, M.: Eco: Efficient convolution operators for tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 6638–6646
34. Long, C., Haizhou, A., Zijie, Z., Chong, S.: Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In: ICME. Volume 5. (2018) 8
35. Dai, J., Li, Y., He, K., Sun, J.: R-fcn: Object detection via region-based fully convolutional networks. In: Advances in neural information processing systems. (2016) 379–387
36. Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K.: Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size. arXiv preprint arXiv:1602.07360 (2016)
37. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. (2015) 91–99
38. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 770–778
39. Li, B., Yan, J., Wu, W., Zhu, Z., Hu, X.: High performance visual tracking with siamese region proposal network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 8971–8980
40. Zhu, Z., Wang, Q., Li, B., Wu, W., Yan, J., Hu, W.: Distractor-aware siamese networks for visual object tracking. In: Proceedings of the European Conference on Computer Vision (ECCV). (2018) 101–117
41. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. (2012) 1097–1105
42. Hamid Rezatofighi, S., Milan, A., Zhang, Z., Shi, Q., Dick, A., Reid, I.: Joint probabilistic data association revisited. In: Proceedings of the IEEE international conference on computer vision. (2015) 3047–3055
43. Shi, X., Ling, H., Pang, Y., Hu, W., Chu, P., Xing, J.: Rank-1 tensor approximation for high-order association in multi-target tracking. International Journal of Computer Vision (2019) 1–21
44. Keuper, M., Tang, S., Zhongjie, Y., Andres, B., Brox, T., Schiele, B.: A multi-cut formulation for joint segmentation and tracking of multiple objects. arXiv preprint arXiv:1607.06317 (2016)
45. Choi, W.: Near-online multi-target tracking with aggregated local flow descriptor. In: Proceedings of the IEEE international conference on computer vision. (2015) 3029–3037
46. Tang, S., Andriluka, M., Andres, B., Schiele, B.: Multiple people tracking by lifted multicut and person re-identification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 3539–3548
47. Wang, G., Wang, Y., Zhang, H., Gu, R., Hwang, J.N.: Exploit the connectivity: Multi-object tracking with trackletnet. In: Proceedings of the 27th ACM International Conference on Multimedia. (2019) 482–490
48. Ma, L., Tang, S., Black, M.J., Van Gool, L.: Customized multi-person tracker. In: Asian Conference on Computer Vision, Springer (2018) 612–628
49. Chen, L., Ai, H., Chen, R., Zhuang, Z.: Aggregate tracklet appearance features for multi-object tracking. IEEE Signal Processing Letters **26** (2019) 1613–1617

50. Kieritz, H., Becker, S., Hübner, W., Arens, M.: Online multi-person tracking using integral channel features. In: 2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), IEEE (2016) 122–130
51. Bochinski, E., Eiselein, V., Sikora, T.: High-speed tracking-by-detection without using image information. In: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), IEEE (2017) 1–6
52. Chen, J., Sheng, H., Zhang, Y., Xiong, Z.: Enhancing detection model for multiple hypothesis tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. (2017) 18–27
53. Sheng, H., Chen, J., Zhang, Y., Ke, W., Xiong, Z., Yu, J.: Iterative multiple hypothesis tracking with tracklet-level association. IEEE Transactions on Circuits and Systems for Video Technology (2018)
54. Keuper, M., Tang, S., Andres, B., Brox, T., Schiele, B.: Motion segmentation & multiple object tracking by correlation co-clustering. IEEE transactions on pattern analysis and machine intelligence (2018)
55. Henschel, R., Leal-Taixe, L., Cremers, D., Rosenhahn, B.: Fusion of head and full-body detectors for multi-object tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. (2018) 1428–1437
56. Fu, Z., Feng, P., Naqvi, S.M., Chambers, J.A.: Particle phd filter based multi-target tracking using discriminative group-structured dictionary learning. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE (2017) 4376–4380
57. Yoon, Y.c., Boragule, A., Song, Y.m., Yoon, K., Jeon, M.: Online multi-object tracking with historical appearance matching and scene adaptive detection filtering. In: 2018 15th IEEE International conference on advanced video and signal based surveillance (AVSS), IEEE (2018) 1–6
58. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: the clear mot metrics. Journal on Image and Video Processing **2008** (2008) 1