Beyond MOT: Semantic Multi-Object Tracking

Yunhao Li[1,2], Qin Li[1], Hao Wang[2], Xue Ma[1], Jiali Yao[3], Shaohua Dong[4],
Heng Fan[4†], and Libo Zhang[1,2,3†♯]

[1] Institute of Software Chinese Academy of Sciences
[2] University of Chinese Academy of Sciences
[3] Hangzhou Institute for Advanced Study, University of Chinese Academy of Sciences
[4] Department of Computer Science & Engineering, University of North Texas

Abstract. Current multi-object tracking (MOT) aims to predict trajectories of targets (*i.e.*, "*where*") in videos. Yet, knowing merely "*where*" is insufficient in many crucial applications. In comparison, semantic understanding such as fine-grained behaviors, interactions, and overall summarized captions (*i.e.*, "*what*") from videos, associated with "*where*", is highly-desired for comprehensive video analysis. Thus motivated, we introduce *Semantic Multi-Object Tracking* (SMOT), that aims to estimate object trajectories and meanwhile understand semantic details of associated trajectories including instance captions, instance interactions, and overall video captions, integrating "*where*" and "*what*" for tracking. In order to foster the exploration of SMOT, we propose BenSMOT, a large-scale Benchmark for Semantic MOT. Specifically, BenSMOT comprises 3,292 videos with 151K frames, covering various scenarios for semantic tracking of humans. BenSMOT provides annotations for the trajectories of targets, along with associated instance captions in natural language, instance interactions, and overall caption for each video sequence. To our best knowledge, BenSMOT is the *first* publicly available benchmark for SMOT. Besides, to encourage future research, we present a novel tracker named SMOTer, which is specially designed and end-to-end trained for SMOT, showing promising performance. By releasing BenSMOT, we expect to go beyond conventional MOT by predicting "*where*" and "*what*" for SMOT, opening up a new direction in tracking for video understanding. We will release BenSMOT and SMOTer at `https://github.com/Nathan-Li123/SMOTer`.
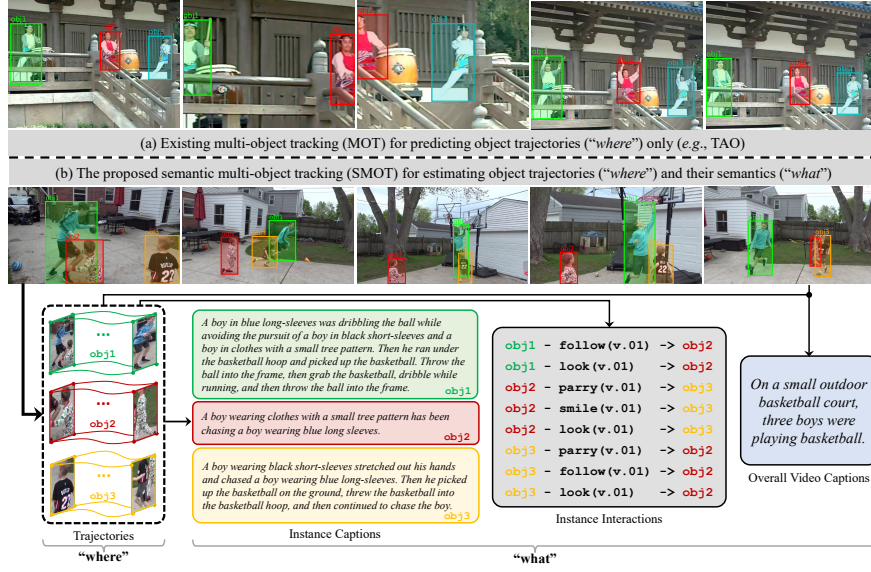
Keywords: Semantic Multi-Object Tracking (SMOT) · Benchmark

1 Introduction

Multi-object tracking (MOT) is a fundamental problem in computer vision with many applications such as autonomous driving, robotics, and visual surveillance. Current MOT tasks (*e.g.*, [9,18,33,38,48]) usually focus on predicting trajectories of the targets from videos, *i.e.*, answering the question of "*where are the targets*". Despite considerable advancements in deep learning era (*e.g.*, [17,32,44,55,60]),

---

(a) Existing multi-object tracking (MOT) for predicting object trajectories ("*where*") only (*e.g.*, TAO)

(b) The proposed semantic multi-object tracking (SMOT) for estimating object trajectories ("*where*") and their semantics ("*what*")

A boy in blue long-sleeves was dribbling the ball while avoiding the pursuit of a boy in black short-sleeves and a boy in clothes with a small tree pattern. Then he ran under the basketball hoop and picked up the basketball. Throw the ball into the frame, then grab the basketball, dribble while running, and then throw the ball into the frame.   obj1

A boy wearing clothes with a small tree pattern has been chasing a boy wearing blue long sleeves.   obj2

A boy wearing black short-sleeves stretched out his hands and chased a boy wearing blue long-sleeves. Then he picked up the basketball on the ground, threw the basketball into the basketball hoop, and then continued to chase the boy.   obj3

```
obj1 - follow(v.01)  -> obj2
obj1 - look(v.01)    -> obj2
obj2 - parry(v.01)   -> obj3
obj2 - smile(v.01)   -> obj3
obj2 - look(v.01)    -> obj3
obj3 - parry(v.01)   -> obj2
obj3 - follow(v.01)  -> obj2
obj3 - look(v.01)    -> obj2
```

On a small outdoor basketball court, three boys were playing basketball.

Overall Video Captions

Trajectories   Instance Captions   Instance Interactions

"*where*"   "*what*"

Fig. 1: Existing multi-object tracking (MOT) focusing on predicting trajectories only (see (a)) and our semantic multi-object tracking (SMOT) aiming at estimating trajectories and understanding their semantics (see (b)). Best viewed in color for all figures.

knowing merely "*where*" in existing MOT is *insufficient* for video understanding in many important applications. For example, besides trajectories, knowing more trajectory-associated *semantic* details of the objects, such as the behaviors and their interactions with the surroundings (*i.e.*, "*what*"), is greatly beneficial for visual surveillance and robotics, which grows the interest in expanding modern MOT (*i.e.*, addressing "*where*") with trajectory-associated semantic analysis (*i.e.*, understanding "*what*") for more comprehensive video understanding.

Thus motivated, in this work we introduce a new type of MOT task, dubbed *Semantic Multi-Object Tracking* (**SMOT**)[5], which aims to expand scope of MOT beyond merely predicting trajectories to capture rich semantic details of objects in videos. Specifically, besides trajectory estimation, SMOT comprises three additional trajectory-associated semantic understanding tasks, including *instance captioning*, *instance interaction recognition*, and *video captioning*, as illustrated in Fig. 1. In particular, instance captioning aims at describing objects and their behaviors in human language, answering "*what are the objects doing*"; instance interaction recognition is to capture relations between objects, answering "*what are the relations between objects*"; video captioning provides the overall scenario understanding based on trajectories, answering "*what is happening*". Note that, all these three additional tasks are associated with object trajectories in videos. Ultimately, the goal of SMOT is to integrate "*where*" (*i.e.*, object trajectories)

---

[5] Here by "semantic", we emphasize high-level trajectory-based activity understanding in videos in the context of tracking, instead of category as in semantic segmentation.

and "*what*" (*i.e.*, trajectory-associated semantics) for video understanding, going beyond MOT predicting only trajectories (see again Fig. 1). Compared to conventional MOT, our SMOT is a multimodal task across vision and language, which is more challenging yet practical. It is worthy to notice that, SMOT is a natural extension of MOT. Despite sharing some similarities with video captioning [23], SMOT *differs* in that it aims at language understanding of dense target trajectories from videos in the context of instance-level tracking.

To foster study of SMOT, we propose **BenSMOT**, a large-scale Benchmark for Semantic MOT. Specifically, BenSMOT contains 3,292 video sequences with 151K frames, captured from more than 40 diverse daily-life scenarios for **human-centric** semantic tracking. In BenSMOT, there exist more than 7.8K instances, labeled with over 335K bounding boxes for trajectory prediction. For each instance, we provide fine-grained human language to describe the behaviors for instance captioning, resulting in 7.8K sentences. In addition, to capture interactions between objects, BenSMOT provides more than 14K interaction annotations from a rich collection of 335 kinds of interactions. Moreover, for each sequence, a summary text is provided for overall understanding of instances and backgrounds, leading to 3,292 video caption texts in total in BenSMOT. To ensure the high quality of BenSMOT, annotations are manually labeled with careful inspection and refinement. *To our best knowledge*, BenSMOT is the **first** publicly available benchmark dedicated to SMOT. By releasing BenSMOT, we expect it to serve as a platform for advancing the research on SMOT.

Furthermore, in order to facilitate the development of SMOT algorithms on BenSMOT, we present a simple yet effective tracker dubbed **SMOTer**. Specifically, SMOTer is built upon a multi-object tracking model [53]. After generating object trajectories, we extract features for each trajectory and design additional prediction heads for semantic understanding of instance captioning, interaction recognition, and video captioning. Note, SMOTer is *not* a simple offline assembly of models for different tasks. Instead, it is specially designed for SMOT and trained in an end-to-end manner for predicting trajectories and understanding their semantics in videos. Despite simplicity, SMOTer shows promising performance for SMOT, and outperforms the offline-combination strategies, evidencing its effectiveness. We expect it to provide a reference for future research.

We notice there exists a concurrent work combining multi-object tracking and trajectory captioning for dense video object captioning [58]. Compared with [58], our work differs in three aspects. First, *task-wise*, besides trajectory estimation and captioning, the SMOT in this work contains instance interaction and overall video captioning, providing more semantic details. Second, *dataset-wise*, we introduce a large-scale benchmark, BenSMOT, that is dedicated for SMOT and supports complete end-to-end model training, while the work of [58] borrows datasets from different tasks, which results in disjoint training of the model on different tasks and may thus degrade performance. Third, *model-wise*, owing to BenSMOT, we propose an end-to-end algorithm that demonstrates promising results on SMOT, while the approach in [58] is disjointly learned and does not support predicting instance interactions and overall video captioning.

In summary, we make the following contributions: ♠ We introduce semantic multi-object tracking (SMOT), a new tracking paradigm that expands existing MOT task by integrating "*where*" and "*what*"; ♥ We present BenSMOT, a large-scale dataset with 3,292 videos and rich annotations for SMOT; ♣ We propose SMOTer, a simple but effective tracker to facilitate future research of SMOT; ♦ We show that SMOTer achieves promising performance for SMOT and conduct in-depth analysis on it to provide guidance for future algorithm design.

2   Related Works

MOT Benchmarks. Benchmarks have played a crucial role in facilitating the development of MOT. PETS2009 [16] is one of the earliest benchmarks for multi-pedestrian tracking. Later, the popular MOT Challenge [10, 24, 33] has been introduced with more crowded videos and greatly advanced MOT. KITTI [18] and BDD100K [48] are specifically designed for object tracking in autonomous driving. ImageNet-Vid [11] provides trajectory annotations for 30 categories across over 1,000 videos, and TAO [9] further expands object classes to 833 for generic multi-object tracking. To foster MOT in specific dancing and sport scenarios, DanceTrack [38] and SportsMOT [8] have been presented for dancer and player tracking. AnimalTrack [52] focuses on tracking various animals in wild scenes. Moreover, UAVDT [12] and VisDrone [61] provide platforms for tracking targets with drones. ***Different from*** the above MOT benchmarks for object trajectory prediction only ("*where*"), BenSMOT is specially developed for the new task of SMOT to serve as a platform for simultaneous estimation of target trajectories and trajectory-associated semantic understanding. To this end, it offers rich annotations of not only object trajectories as in existing standard MOT datasets ("*where*") but also semantic details of objects ("*what*") such as instance trajectory captions, interactions and overall video captions (see Fig. 1 again).

MOT Algorithms. MOT algorithms have seen great progress in recent years. One of the popular paradigms for multi-object tracking is the so-called *Tracking-by-Detection*. This paradigm involves initial object detection followed by association based on these detections, forming the basis for many representative methods [5,13,21,31,43,53]. In this case, MOT methods typically improve their performance by enhancing both detection and matching effectiveness. In addition, another prevalent paradigm is the *Joint-Tracking-and-Detection* [20, 44, 45, 47, 59], which integrates tracking and detection steps into a single process, enabling end-to-end training. Recently, Transformer [41] has been introduced into MOT and exhibited remarkable improvements over previous tackers [7,17,32,39,50,55,60]. Our SMOTer is related to but ***different than*** existing MOT methods. Specifically, going beyond merely predicting object trajectories, SMOTer also aims at semantic understanding of trajectories, integrating "*where*" and "*what*".

Video Captioning. Video captioning is a multimodal vision-language task that aims to automatically describe the video content using natural language. Owing to its important applications in video event commentary and human-computer interaction, video captioning has drawn increasing attention in the past decade

with many models proposed (*e.g.*, [23, 27, 36, 37, 46, 57]). Similar to the video captioning task, SMOT utilizes natural language to describe instances and the overall video content. However, the ***difference*** is that our SMOT aims at language comprehension for dense target trajectories in the context of multi-object tracking, which is more challenging due to the requirement of accurate trajectories yet provides more fine-grained information for video understanding.

Visual Relationship Detection. The goal of visual relationship detection task is to identify and comprehend the relationships (usually represented by a triplet of <*subject*, *predicate*, *object*>) between objects from an image (*e.g.*, [25,29,51]). Besides static images, recent works extend visual relationship detection in the video domain (*e.g.*, [6, 28, 56]). Our work is relevant to video visual relationship detection but ***different*** by extending this task target-specific trajectories, and thus more challenging due to complicated multi-object tracking scenarios.

3   The Proposed BenSMOT

3.1   Design Principle

BenSMOT aims at providing a new platform for exploring human-centric SMOT. In construction of BenSMOT, we follow the following rules:

(1) *Dedicated benchmark.* The key motivation of our BenSMOT is to provide a dedicated dataset for semantic multi-object tracking. In current deep learning era, a large number of videos are desired in benchmark construction for training robust tracking models. Considering this, we hope to build a dedicated dataset with more than 3,000 videos for human-centric semantic multi-object tracking.

(2) *Diverse scenarios.* For a dataset, diversity is crucial, in both training and evaluation, for developing general systems. In order to provide a diverse platform for SMOT, in BenSMOT we will include videos from more than 40 various scenes that range from daily life scenarios to dancing scenarios to sport scenarios.

(3) *High-quality annotations.* High-quality annotation is crucial for a benchmark in both training and assessing models. In BenSMOT, we ensure its high quality by providing manual annotations for each sequence including object trajectories, instance captions, instance interactions and overall video captions. In this process, multi-round inspections and refinements will be carried out.

3.2   Data Acquisition

BenSMOT focuses on predicting the target trajectories and meanwhile understanding their semantics by instance captioning, interaction recognition, and overall video captioning. To this end, the video sequences in BenSMOT are desired to involve diverse multi-person activities and interactions, which is different from videos in existing benchmarks to some extent. Specifically, we first identify 47 common scenarios, involving different activities ranging from daily-life talk and play to different sports, that are suitable for tracking by drawing inspiration from [22]. Due to limited space, please refer to **supplementary material** for

Table 1: Summary of BenSMOT and its comparison with popular multi-object tracking benchmarks. "n/a" denotes that data is not available.
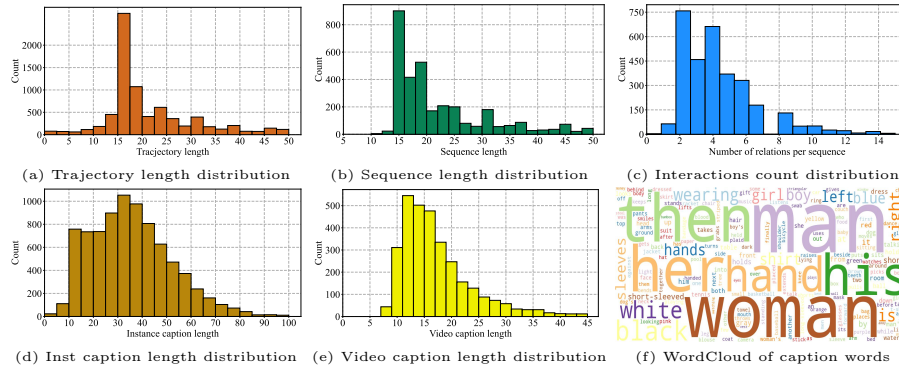
	KITTI [18]	MOT17 [33]	MOT20 [10]	BDD100k [48]	TAO [9]	GMOT-40 [1]	DanceTrack [38]	SportsMOT [8]	BenSMOT (ours)
Videos	50	14	8	1,600	2,907	40	100	240	3,292
Min. length (s)	n/a	17.0	17.0	40.0	n/a	3.0	n/a	n/a	1.5
Avg. length (s)	10.0	33.0	66.8	40.0	36.8	8.9	52.9	n/a	22.9
Max. length (s)	n/a	85.0	133.0	40.0	n/a	24.2	n/a	n/a	116.0
Total length (s)	498	463	535	640,000	106,978	356	5,292	6015	75,499
Total tracks	2,600	1.3K	3.83K	131K	17,287	2,026	990	3,401	7,792
Total boxes	80K	300K	2,102K	3,300K	333K	256K	n/a	1,629K	335K
Total frames	15K	11K	13K	318K	2,674K	9K	106K	150K	151K
Instance Captions	✗	✗	✗	✗	✗	✗	✗	✗	7,792
Instance Interactions	✗	✗	✗	✗	✗	✗	✗	✗	14K
Video Summaries	✗	✗	✗	✗	✗	✗	✗	✗	3,292

detailed scenarios in BenSMOT. Afterwards, we search for raw sequences for each scenario under the `Creative Commons License` from YouTube, the largest and most popular video platform with massive real-world videos. Please notice, for each raw video, instead of using the whole sequence, we usually choose a suitable clip for our semantic multi-object tracking task.

Eventually, we compile a dataset for SMOT by including 3,292 videos with 151K frames. BenSMOT has an average video length of 23 seconds, with our current focus in this work not being on long-term SMOT. The total number of instance tracks is 7.8K, annotated with 335K bounding boxes for tracking. Besides bounding boxes, we provide 7.8K instance captions, 14K interactions, and 3,292 video summaries for trajectory-associated semantic understanding (how to annotate will be described in the following). To our best knowledge, BenSMOT by far is the first publicly available benchmark dedicated for SMOT. Tab. 1 summarizes our BenSMOT and compares it with popular MOT benchmarks.

3.3   Data Annotation

In order to meet the requirements of SMOT, BenSMOT provides four types of annotations, including bounding box, instance caption, instance interaction, and the overall video caption. Specifically, for each trajectory in videos, similar to standard MOT benchmarks (*e.g.*, [9, 33]), we provide axis-aligned bounding boxes to indicate its spatial positions in the videos. For instance captioning, we label each trajectory with a precise sentence in natural language to describe the detailed behavior of the associated target. For interaction recognition, we first collect a set of 335 interactions (*i.e.*, verbs), among them 327 from WordNet [15]. Note that, in our interaction set, the same verb may have different meanings and thus is assigned to different interaction types, *e.g.*, `hold(v.01)` for "being the physical support of someone" and `hold(v.02)` for "someone in one's hands or arm", for precise relationship description. Due to limited space, we refer reader to **supplementary material** for the full set of interaction types. After this, we provide an directed interaction label, represented with a triplet of $<subject$, *predicate*, *object*$>$ (see Fig. 1 (b)), between two trajectories if associated objects

(a) Trajectory length distribution   (b) Sequence length distribution   (c) Interactions count distribution

(d) Inst caption length distribution   (e) Video caption length distribution   (f) WordCloud of caption words

Fig. 2: Representative statistics on BenSMOT, including distributions of target trajectory length (in seconds) in (a), sequence length (in seconds) in (b), number of interactions per sequence in (c), instance caption length in (d), video caption length in (e), and wordcloud of all caption words with prepositions excluded in (f).

are interacting. Finally, for the overall video captioning, we utilize a concise text summary to describe object trajectories in the video from a global perspective.

To ensure the high-quality annotations, we adopt a multi-round mechanism. Specifically, each video will be first manually annotated by a few volunteers who are familiar with the task and an expert working on related problems. After this initial round, all annotations of the video will be sent to a validation team former by more than two experts for inspection. If initial annotations are not unanimously agreed by the experts, they will be returned to the original labeling team for refinement using the feedback from the validation team. We repeat this procedure for each video until the annotations of all video sequences in BenSMOT are completed. An annotation example in our BenSMOT is shown in Fig. 1 (b), and more can be found in **supplementary material**.

Statistics of annotation. In order to better understand BenSMOT, we show some representative statics of annotations in Fig. 2. Specifically, we demonstrate the distributions of object trajectory length, video length, number of instance interactions, instance caption length, video caption length, and wordcloud. From Fig. 2 (c), it is worth noting that different targets in videos have frequent interactions. In addition, for the instance captions in Fig. 2 (d), we provide relatively longer textual descriptions, which allows precise understanding.

3.4   Dataset Split and Evaluation Metric

Training/Test Split. BenSMOT contains 3,292 videos, captured from 47 different scenarios. Within each scenario, we use 70% of the sequences for training, and the rest 30% for test. During dataset split, we try to keep the distributions of training and test sets similar. Eventfully, the training set of BenSMOT comprises 2,284 sequences with 104K frames, and the test set consists of 1,008 videos with 47K frames. Please see **supplementary material** for more split details.
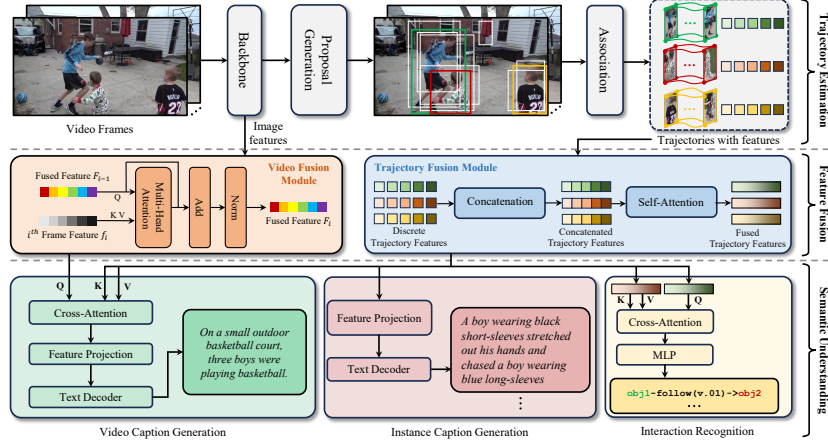
Fig. 3: Illustration of SMOTer, which contains three components of trajectory estimation for tracking, feature fusion, and trajectory-associated semantic understanding.

Evaluation Metric. We apply multiple metrics on BenSMOT for evaluation. Specifically, to assess the performance in object trajectory estimation, we employ higher order tracking accuracy (HOTA), association accuracy (AssA), detection accuracy (DetA), and localization accuracy (LocA) by following [30], CLEAR metrics [3] including multiple object tracking accuracy (MOTA), false positives (FP), false negatives (FN), and ID switches (IDs), and ID metrics [35] containing identification precision (IDP), identification recall (IDR) and related F1 score (IDF1). For instance and video captioning, we follow existing methods (*e.g.*, [27, 37]) and employ BLEU [34], ROUGE [26], METEOR [2], and CIDEr [42] for thorough evaluation. Lastly, to assess performance of interaction recognition, we leverage classic metrics such as Precision (Prcn), Recall (Rcll), and F1, which are widely used in the context of classification tasks.

4   Methodology: A New Baseline for SMOT

Overview. To encourage development of SMOT algorithms, we propose a simple yet effective tracker on BenSMOT, dubbed SMOTer. As illustrated in Fig. 3, SMOTer logically consists of three key components. The first component, as detailed in Sec 4.1, is dedicated to object trajectory estimation for tracking, which is the basis for SMOT. Then, the second component, as explained in Sec 4.2, focuses on aggregating features from each frame into the overall video feature and meanwhile merging features of individual objects into target trajectory features using two distinct fusion modules, for subsequent semantic understanding. Finally, the third component, as described in Sec 4.3, aims to predict trajectory-associated semantic details in the video, including instance captions, instance interactions, and overall video caption. Please notice that, our SMOTer is specially designed for SMOT and trained in an end-to-end manner, as in Sec 4.4.

4.1   Target Trajectory Estimation for Tracking

Give a video with N frames, SMOTer first uses a CNN backbone [49] to extract their features $\{f_i\}_{i=1}^N$, where f_i denotes the feature of the ith frame. Then, an initial set of proposals is generated using proposal generator of a popular detection architecture [14] with threshold filtering by τ_p. Afterwards, all reserved proposals are send to the association module (BYTE [53] is used in SMOTer for association) to obtain object trajectories. Finally, leveraging target coordinates within the trajectories and image features, target features are acquired for each target in trajectories through RoI Pooling [19]. Specifically, given feature f_i of the ith frame and target coordinates B_j^i corresponding to the jth object trajectory in this frame, we extract the corresponding target feature $t_j^i = \texttt{RoIPooling}(f_i, B_j^i)$ and apply it in subsequent feature fusion for semantic understanding.

In summary, at this stage, SMOTer predicts object trajectories for tracking in a given video and extracts image and target features for subsequent stages.

4.2   Feature Fusion

Given image and target features from previous stage in Sec. 4.1, we design two feature fusion modules, including attention-based Video Fusion Module (VFM) and concatenation-based Trajectory Fusion Module (TFM). VFM aims to merge image-level features into overall video features while TFM focuses on integrating target features from each frame into target trajectory features.

Video Fusion Module. VFM takes features $\{f_i\}_{i=1}^N$ of N frames as input and sequentially feeds them into a cross-attention module for video feature fusion. In specific, the fused video feature F_i up to the ith ($i > 1$) frame is computed as

$$F_i = \texttt{CA}(F_{i-1}, f_i)   i > 1 \tag{1}$$

where $\texttt{CA}(\mathbf{z}, \mathbf{u})$ denotes cross-attention with $\mathbf{z}$ generating query and $\mathbf{u}$ key/value as in [41]. For the first frame ($i = 1$), $F_1 = f_1$. Finally, with the help of VFM, we can generate the overall fused video feature $\tilde{F} = F_N$.

Trajectory Fusion Module. Unlike VFM, TFM is to integrate multiple target features belonging to the same object trajectory, which are distributed across different frames, into a single trajectory feature. Concretely, given t_i^j for object feature of trajectory j in frame i, we perform average pooing to yield a relatively simple 2D feature $\hat{t}_i^j = \texttt{AvgPooling}(t_i^j)$, where $\hat{t}_i^j \in \mathbb{R}^{4 \times d}$ with dimension $d = 256$. Then, for the jth object trajectory, we first concatenate its discrete features into an integrated one $\hat{T}_j = \texttt{Concat}(\hat{t}_1^j, \hat{t}_2^j, \cdots, \hat{t}_{N_j}^j)$, where N_j denotes trajectory length. After that, self-attention is used to enhance trajectory feature as follows

$$\tilde{T}_j = \texttt{SA}(\hat{T}_j) \tag{2}$$

where $\texttt{SA}(\mathbf{z})$ denotes self-attention with $\mathbf{z}$ generating query/key/value as in [41], and $\tilde{T}_j$ is the enhanced feature for trajectory j.

4.3   Semantic Understanding

In the third phase, SMOTer applies the fused video feature $\tilde{F}$ and trajectory features $\tilde{T} = \{\tilde{T_1}, \tilde{T_2}, ..., \tilde{T_M}\}$, where M denotes the number of trajectories in the video, to tackle trajectory-associated semantic understanding tasks in SMOT, *i.e.*, video captioning, instance captioning, and interaction recognition.

Video captioning. Unlike current video captioning, in SMOTer we incorporate trajectory features into describing video content and allow more comprehensive human-centric caption generation. Specifically, SMOTer sequentially injects trajectory features $\tilde{T}$ into video feature $\tilde{F}$ with cross-attention and then predicts captioning result R_{Vid} via a linear project layer and a text decoder, as follows

$$R_{\text{Vid}} = \text{Dec}_{\text{Vid}}(\text{Linear}(\text{CA}(\tilde{F}, \tilde{T}))) \tag{3}$$

where $\text{Dec}_{\text{Vid}}(\cdot)$ is a video text decoder for video captioning (please kindly refer to **supplementary material** for details), and $\text{Linear}(\cdot)$ is a linear projection.

Instance captioning. Different from video captioning, for instance captioning, we directly generate the results based on the target trajectory features. Specifically, given the feature $\tilde{T_j}$ for the jth ($1 \leq j \leq M$) trajectory, we apply a linear projection layer followed by the designed text decoder to predict the captioning result R_{Ins}^j of each trajectory, depicted as follows,

$$R_{\text{Ins}}^j = \text{Dec}_{\text{Ins}}(\text{Linear}(\tilde{T_j})) \quad (1 \leq j \leq M) \tag{4}$$

where $\text{Dec}_{\text{Ins}}(\cdot)$ is instance text decoder [46] for instance captioning. Please refer to **supplementary material** for more details because of space limitation.

Interaction recognition. Besides captioning, SMOTer predicts the interaction between trajectories, which is crucial for video understanding. Particularly, in SMOTer we concentrate solely on interactions between two trajectories, as our current focus doesn't extend to understanding interactions involving three or more targets, leaving it for future research. More specifically, for two trajectories j and k, we assume the jth trajectory in a video sequence is the active trajectory and the kth trajectory the passive trajectory. Then, we first fuse the features of active and passive trajectories using cross-attention and then predict the interaction result $R_{\text{Int}}^{j,k}$ using multi-layer perception [40], as follows,

$$R_{\text{Int}}^{j,k} = \text{MLP}(\text{CA}(\tilde{T_j}, \tilde{T_k})), \tag{5}$$

where $\text{MLP}(\cdot)$ denotes the multi-layer perception. Please note, interchanging the active and passive trajectories may yield disparate outcomes, *i.e.*, $R_{\text{Int}}^{j,k}$ and $R_{\text{Int}}^{k,j}$ may exhibit dissimilar characteristics, as indicated by the formula above.

4.4   End-to-end Training

End-to-end training of SMOTer is not easy due to conflict between two types of training requirements, *i.e.*, the trajectory estimation is usually trained frame-by-frame, while the captioning tasks, and interaction recognition are often trained

at the entire video level, resulting in two different losses in training. In addition, due to the frame-by-frame training nature of trajectory estimation for tracking, the complete sequence is hard to be input at once during training, preventing the model from obtaining full trajectories thus hindering subsequent caption generation and interaction prediction. To address these issues, we draw inspiration from [60] and adopt a strategy of completing full trajectory association after detection on the entire video. Specifically, we first compute detection loss after each frame input and then perform trajectory association after the entire sequence is detected. Subsequently, with complete trajectories, we compute losses for trajectory-associated instance captioning, interaction recognition, and video captioning tasks, realizing end-to-end training of SMOTer. For the detailed training loss of SMOTer, please kindly refer to the **supplementary material**.

5   Experiments

Experimental setup. We conduct experiments with 4 Nvidia Tesla V100 GPUs (32GB). The batch size is 1 per GPU. Each batch has a video clip with multiple frames. We use the AdamW optimizer with an initial learning rate of 5.0×10^{-4}. During training, we filter out proposals with scores lower than the threshold $\tau_p = 0.3$. For lost tracklets, we retain them for 30 frames in case they reappear. Due to memory constraints and limited accuracy of trajectory estimation in early training stages, we sample 6-8 frames per sequence for training detection model and only compute tracking related losses in the first 20K training iterations.

As SMOT is new and no suitable methods exist for comparison with SMOTer, we logically divide SMOT into two components: trajectory estimation for tracking and semantic understanding (including instance/video captioning, and interaction recognition). Subsequently, we devise a series of two-stage models based on existing MOT frameworks. Leveraging several state-of-the-art and classic MOT models such as SORT [4], DeepSORT [43], OC-SORT [5], ByteTrack [53], TransTrack [39], MOTR [50], and MOTRv2 [55] as foundations, we integrate feature fusion and semantic generation functionalities akin to SMOTer and conduct comparative experiments on BenSMOT. To ensure fair experiments, we employ the same training framework for comparative experiments of two-stage approaches. Additionally, for models based on two-stage MOT methods like SORT, we use the same detector [14] as SMOTer during training and evaluation.

5.1   Comparison on Trajectory Estimation for Tracking.

We first compare SMOTer with other MOT methods for trajectory estimation on BenSMOT. Specifically, SMOTer is fully end-to-end trained, while other two-stage MOT models are trained with their own strategies, concentrating solely on tracking. As depicted in Tab. 2, despite being at a certain degree of natural disadvantage, our SMOTer achieves comparable or even superior performance in tracking compared to state-of-the-art MOT models. SMOTer consistently

Table 2: Comparison between SMOTer and two-stage MOT methods regarding tracking performance on BenSMOT. The best two results are shown in red and blue fonts.

Method	HOTA↑	AssA↑	DetA↑	LocA↑	MOTA↑	FN↓	FP↓	IDs↓	IDR↑	IDP↑	IDF1↑
SORT [4]	48.49	38.95	60.91	87.50	53.58	24001	5105	13875	60.85	48.43	53.93
DeepSORT [43]	50.12	40.23	61.45	87.67	54.29	22890	5540	11278	62.10	51.11	56.76
OC-SORT [5]	51.00	41.42	63.31	87.61	55.19	21061	5388	15049	63.92	53.10	58.01
ByteTrack [53]	68.84	71.15	67.10	85.15	73.87	15419	7070	1712	82.25	74.83	78.37
TransTrack [39]	71.31	73.34	69.67	91.31	74.08	20124	4420	2530	85.63	72.75	78.67
MOTR [50]	66.10	73.12	55.14	86.30	45.19	31297	11178	617	72.39	70.12	68.97
MOTRv2 [55]	65.28	76.82	51.30	86.09	45.52	40765	20923	430	78.47	65.51	70.76
SMOTer (ours)	71.98	73.71	70.79	87.11	77.71	12534	6388	1702	83.82	77.97	80.65

Table 3: Comparison of SMOTer against two-stage methods based on MOT models regarding semantic understanding. The best two results are highlighted in red and blue.

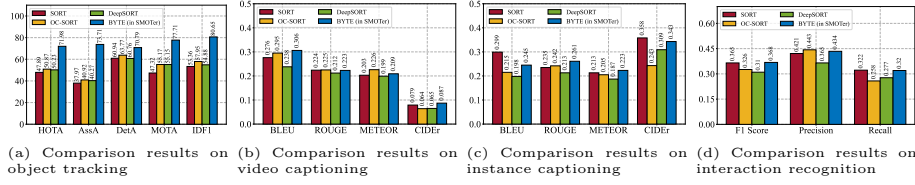|	Video Caption				Instance Caption				Interaction		
Method	BLEU↑	ROUGE↑	METEOR↑	CIDEr↑	BLEU↑	ROUGE↑	METEOR↑	CIDEr↑	Prec↑	Rcll↑	F1↑
SORT [4]	0.245	0.224	0.202	0.298	0.233	0.245	0.208	0.056	0.363	0.259	0.302
DeepSORT [43]	0.198	0.213	0.187	0.309	0.238	0.212	0.199	0.065	0.365	0.277	0.310
OC-SORT [5]	0.231	0.252	0.215	0.242	0.270	0.205	0.180	0.033	0.384	0.291	0.331
ByteTrack [53]	0.224	0.225	0.212	0.266	0.304	0.242	0.224	0.064	0.443	0.258	0.326
TransTrack [39]	0.247	0.248	0.209	0.269	0.283	0.219	0.201	0.074	0.406	0.311	0.376
MOTR [50]	0.187	0.254	0.203	0.244	0.230	0.209	0.182	0.061	0.425	0.314	0.354
MOTRv2 [55]	0.217	0.258	0.219	0.248	0.238	0.241	0.204	0.059	0.313	0.395	0.349
SMOTer (ours)	0.245	0.261	0.223	0.343	0.306	0.223	0.209	0.087	0.434	0.320	0.368

Table 4: Ablation experiments for evaluating various feature fusion strategies using results of three additional tasks. The best result is highlighted in red.

	VFM (video fusion)				TFM (trajectory fusion)						
	BLEU↑	ROUGE↑	METEOR↑	CIDEr↑	BLEU↑	ROUGE↑	METEOR↑	CIDEr↑	Prcn↑	Rcll↑	F1↑
Attention-based	0.245	0.261	0.223	0.343	0.303	0.218	0.208	0.072	0.406	0.347	0.364
MLP-based	0.251	0.223	0.200	0.220	0.337	0.219	0.205	0.072	0.380	0.252	0.303
Concatenation-based	0.230	0.213	0.215	0.365	0.306	0.223	0.209	0.087	0.434	0.320	0.368
Addition-based	0.140	0.157	0.104	0.127	0.182	0.137	0.114	0.041	0.228	0.163	0.205

achieves top-tier performance across nearly all metrics, demonstrating impressive results in key metrics. For instance, it achieves a notable 71.98% in HOTA, 77.71% in MOTA, and 80.65% in IDF1. This suggests that subsequent training tasks do not degrade tracking performance, validating effectiveness of our end-to-end training strategy. In fact, by comparing with the standard ByteTrack, we observe that although SMOTer adopts a highly similar structure to ByteTrack in tracking, it achieves superior performance, showing +3.15% in HOTA and +3.74% in MOTA. Moreover, even in metrics where it does not excel, such as FP and LocA, SMOTer exhibits significant improvements compared to ByteTrack, implying that semantic understanding tasks can reciprocally aid in tracking.

5.2   Comparison on Semantic Understanding

Besides tracking, we compare SMOTer with a set of two-stage models on their semantic understanding capabilities. Similarly, SMOTer undergoes complete end-to-end training. However, other two-stage models first train trackers on their own to acquire target trajectories and then use these trajectories to separately train the semantic understanding tasks (with the same feature fusion as SMOTer).

(a) Comparison results on object tracking (b) Comparison results on video captioning (c) Comparison results on instance captioning (d) Comparison results on interaction recognition

Fig. 4: Ablation experiments for different association strategies, including comparison results on object tracking in image (a), on video captioning in image (b), on instance captioning in image (c), and on interaction recognition in image (d).

From Tab. 3, we can see SMOTer achieves highly favorable results on almost all metrics. Particularly, compared to the baseline method ByteTrack, SMOTer surpasses it on most metrics, including +2.1% BLEU and +7.7% CIDEr on video captioning, +2.3% CIDEr on instance captioning, and +4.2% F1 score on interaction recognition. More results can be seen in **supplementary material**.

5.3   Ablation Study

Impact of different feature fusion. Feature fusion is a crucial component of SMOTer for generating overall video and enhanced trajectory features. In this work, we study four types of feature fusion strategies for VFM and TFM, including attention-based fusion (using cross-attention), MLP-based fusion (using MLP module), concatenation-based fusion, and addition-based fusion. Please refer to **supplementary material** for detailed architectures. To assess the impact for VFM, we compare the performance on video caption generation because the fused video feature from VFM is used for this sub-task, while for TFM, we measure the results on instance captioning and interaction recognition since these two sub-tasks rely on the fused trajectory features. Tab. 4 shows the results of different fusion mechanisms for VFM and TFM. We can see that attention-based fusion works generally better for VFM by achieving best ROUGE score of 0.261 and METEOR score of 0.223 and second best BLEU score of 0.245 and CIDEr score of 0.343 for video captioning. For TFM, concatenation-based fusion shows more superior results by obtaining best scores on ROUGE, METEOR, CIDEr for instance captioning and on Precision and F1 for interaction recognition.

Analysis on association mechanism. Association is necessary in SMOTer to generate target trajectories. In our method, we leverage the popular BYTE [54] for proposal association. To analyze the impact of different association strategies, we compare other representative manners, including SORT [4], DeepSORT [43], and OC-SORT [5], with the adopted BYTE. Fig. 4 demonstrates the comparison results of association mechanisms in SMOTer (using BYTE) and other approaches on four tasks, including object tracking (Fig. 4 (a)), video captioning (Fig. 4 (b)), instance captioning (Fig. 4 (c)), and interaction recognition (Fig. 4 (d)). From the results in Fig. 4 (a)-(d), we can observe that BYTE used in our SMOTer can achieve the best or competitive performance on different metrics for various tasks, validating its effectiveness for semantic multi-object tracking.

Table 5: Study of different thresholds τ_p. The best result is highlighted in red.

τ_p	Tracking			Video Caption		Instance Caption		Interaction		
	HOTA↑	MOTA↑	IDF1↑	METEOR↑	CIDEr↑	METEOR↑	CIDEr↑	Prcn↑	Rcll↑	F1↑
0.1	70.51	72.30	78.80	0.219	0.256	0.192	0.069	0.378	0.378	0.319
0.2	71.10	73.80	78.90	0.215	0.289	0.199	0.069	0.384	0.365	0.331
0.3	71.98	77.71	80.65	0.224	0.343	0.209	0.087	0.434	0.320	0.368
0.4	71.19	76.36	76.01	0.214	0.330	0.215	0.091	0.420	0.302	0.341
0.5	68.30	74.34	78.90	0.224	0.343	0.216	0.087	0.396	0.291	0.310

Study of threshold τ_p. The parameter τ_p is used to remove low-confidence proposals to ensure the quality of subsequent trajectories, and thus crucial for all tasks in SMOTer. Because of this, we conduct an ablation study for investigating different τ_p, and Tab. 5 displays the results. From Tab. 5, we can observe that, when $\tau_p = 0.3$, SMOTer achieves the overall best performance for different tasks.

5.4 Discussion

Semantic understanding for improving tracking. SMOT aims to go beyond just tracking by incorporating additional semantic understanding tasks including trajectory-associated instance/video captioning and interaction recognition. We observe that these extra goals greatly improve the tracking performance. For example, as shown in Tab. 2, we can see that joint training of different tasks leads to better performance of SMOTer (71.98%/77.71% in HOTA/MOTA) in tracking compared to its baseline ByteTrack (68.84%/73.87 in HOTA/MOTA). The possible reason is that SMOTer needs to comprehensively understand the video content for semantic multi-object tracking, and the learned semantic knowledge helps distinguish different object trajectories.

Challenge in SMOT. One reason why SMOT is challenging lies in the complicated behaviors of targets, often leading to lengthy instance captions. For instance, a basketball player can be dribbling, shooting, and defending in a game. Models often miss some states in understanding. In BenSMOT, the average video length is about 23 seconds, with objects typically present for around 21 seconds, resulting in instance captions averaging over 35 words. This complexity poses a challenge for SMOT and indicates a potential area for future improvement.

6   Conclusion

In this paper, we introduce SMOT to expand the scope of MOT by integrating "*where*" and "*what*". To facilitate the research of SMOT, we propose the large-scale BenSMOT by including 3,292 videos from more than 40 diverse scenarios. To our best knowledge, BenSMOT is the first benchmark dedicated to SMOT. Furthermore, to encourage algorithm development on BenSMOT, we introduce SMOTer, an end-to-end tracker designed for SMOT. Our results exhibit SMOTer surpasses offline-combination strategies, showing efficacy. By presenting BenSMOT and SMOTer, we hope to inspire more future research on SMOT.

References

1. Bai, H., Cheng, W., Chu, P., Liu, J., Zhang, K., Ling, H.: Gmot-40: A benchmark for generic multiple object tracking. In: CVPR (2021)
2. Banerjee, S., Lavie, A.: Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In: ACL Workshop (2005)
3. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: the clear mot metrics. JIVP (2008)
4. Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B.: Simple online and realtime tracking. In: ICIP (2016)
5. Cao, J., Pang, J., Weng, X., Khirodkar, R., Kitani, K.: Observation-centric sort: Rethinking sort for robust multi-object tracking. In: CVPR (2023)
6. Chen, S., Shi, Z., Mettes, P., Snoek, C.G.: Social fabric: Tubelet compositions for video relation detection. In: ICCV (2021)
7. Chu, P., Wang, J., You, Q., Ling, H., Liu, Z.: Transmot: Spatial-temporal graph transformer for multiple object tracking. In: WACV (2023)
8. Cui, Y., Zeng, C., Zhao, X., Yang, Y., Wu, G., Wang, L.: Sportsmot: A large multi-object tracking dataset in multiple sports scenes. ICCV (2023)
9. Dave, A., Khurana, T., Tokmakov, P., Schmid, C., Ramanan, D.: Tao: A large-scale benchmark for tracking any object. In: ECCV (2020)
10. Dendorfer, P., Rezatofighi, H., Milan, A., Shi, J., Cremers, D., Reid, I., Roth, S., Schindler, K., Leal-Taixé, L.: Mot20: A benchmark for multi object tracking in crowded scenes. arXiv (2020)
11. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009)
12. Du, D., Qi, Y., Yu, H., Yang, Y., Duan, K., Li, G., Zhang, W., Huang, Q., Tian, Q.: The unmanned aerial vehicle benchmark: Object detection and tracking. In: ECCV (2018)
13. Du, Y., Zhao, Z., Song, Y., Zhao, Y., Su, F., Gong, T., Meng, H.: Strongsort: Make deepsort great again. TMM (2023)
14. Duan, K., Bai, S., Xie, L., Qi, H., Huang, Q., Tian, Q.: Centernet: Keypoint triplets for object detection. In: ICCV (2019)
15. Fellbaum, C.: WordNet: An electronic lexical database (1998)
16. Ferryman, J., Shahrokni, A.: Pets2009: Dataset and challenge. In: PET Workshop (2009)
17. Gao, R., Wang, L.: Memotr: Long-term memory-augmented transformer for multi-object tracking. In: ICCV (2023)
18. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset (2013)
19. Girshick, R.: Fast r-cnn. In: ICCV (2015)
20. Han, X., Pasquier, T., Bates, A., Mickens, J., Seltzer, M.: Unicorn: Runtime provenance-based detector for advanced persistent threats. ECCV (2020)
21. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
22. Heilbron, F.C., Escorcia, V., Ghanem, B., Niebles, J.C.: Activitynet: A large-scale video benchmark for human activity understanding. In: CVPR (2015)
23. Krishna, R., Hata, K., Ren, F., Fei-Fei, L., Carlos Niebles, J.: Dense-captioning events in videos. In: ICCV (2017)
24. Leal-Taixé, L., Milan, A., Reid, I., Roth, S., Schindler, K.: Motchallenge 2015: Towards a benchmark for multi-target tracking. arXiv (2015)

25. Liang, X., Lee, L., Xing, E.P.: Deep variation-structured reinforcement learning for visual relationship and attribute detection. In: CVPR (2017)
26. Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: ACL (2004)
27. Lin, K., Li, L., Lin, C.C., Ahmed, F., Gan, Z., Liu, Z., Lu, Y., Wang, L.: Swinbert: End-to-end transformers with sparse attention for video captioning. In: CVPR (2022)
28. Liu, C., Jin, Y., Xu, K., Gong, G., Mu, Y.: Beyond short-term snippet: Video relation detection with spatio-temporal global context. In: CVPR (2020)
29. Lu, C., Krishna, R., Bernstein, M., Fei-Fei, L.: Visual relationship detection with language priors. In: ECCV (2016)
30. Luiten, J., Osep, A., Dendorfer, P., Torr, P., Geiger, A., Leal-Taixé, L., Leibe, B.: Hota: A higher order metric for evaluating multi-object tracking. IJCV (2021)
31. Maggiolino, G., Ahmad, A., Cao, J., Kitani, K.: Deep oc-sort: Multi-pedestrian tracking by adaptive re-identification. ICIP (2023)
32. Meinhardt, T., Kirillov, A., Leal-Taixe, L., Feichtenhofer, C.: Trackformer: Multi-object tracking with transformers. In: CVPR (2022)
33. Milan, A., Leal-Taixé, L., Reid, I., Roth, S., Schindler, K.: Mot16: A benchmark for multi-object tracking. arXiv (2016)
34. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: ACL (2002)
35. Ristani, E., Solera, F., Zou, R., Cucchiara, R., Tomasi, C.: Performance measures and a data set for multi-target, multi-camera tracking. In: ECCV (2016)
36. Seo, P.H., Nagrani, A., Arnab, A., Schmid, C.: End-to-end generative pretraining for multimodal video captioning. In: CVPR (2022)
37. Shen, Y., Gu, X., Xu, K., Fan, H., Wen, L., Zhang, L.: Accurate and fast compressed video captioning. In: ICCV (2023)
38. Sun, P., Cao, J., Jiang, Y., Yuan, Z., Bai, S., Kitani, K., Luo, P.: Dancetrack: Multi-object tracking in uniform appearance and diverse motion. In: CVPR (2022)
39. Sun, P., Cao, J., Jiang, Y., Zhang, R., Xie, E., Yuan, Z., Wang, C., Luo, P.: Transtrack: Multiple object tracking with transformer. arXiv (2020)
40. Taud, H., Mas, J.: Multilayer perceptron (mlp). Geomatic approaches for modeling land change scenarios (2018)
41. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. NIPS (2017)
42. Vedantam, R., Lawrence Zitnick, C., Parikh, D.: Cider: Consensus-based image description evaluation. In: CVPR (2015)
43. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. In: ICIP (2017)
44. Yan, B., Jiang, Y., Sun, P., Wang, D., Yuan, Z., Luo, P., Lu, H.: Towards grand unification of object tracking. In: ECCV (2022)
45. Yan, B., Jiang, Y., Wu, J., Wang, D., Luo, P., Yuan, Z., Lu, H.: Universal instance perception as object discovery and retrieval. In: CVPR (2023)
46. Yang, A., Nagrani, A., Seo, P.H., Miech, A., Pont-Tuset, J., Laptev, I., Sivic, J., Schmid, C.: Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In: CVPR (2023)
47. Ye, B., Chang, H., Ma, B., Shan, S., Chen, X.: Joint feature learning and relation modeling for tracking: A one-stream framework. In: ECCV (2022)
48. Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., Darrell, T.: Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In: CVPR (2020)

49. Yu, F., Wang, D., Shelhamer, E., Darrell, T.: Deep layer aggregation. In: CVPR (2018)
50. Zeng, F., Dong, B., Zhang, Y., Wang, T., Zhang, X., Wei, Y.: Motr: End-to-end multiple-object tracking with transformer. In: ECCV (2022)
51. Zhang, H., Kyaw, Z., Chang, S.F., Chua, T.S.: Visual translation embedding network for visual relation detection. In: CVPR (2017)
52. Zhang, L., Gao, J., Xiao, Z., Fan, H.: Animaltrack: A benchmark for multi-animal tracking in the wild. IJCV (2023)
53. Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W., Wang, X.: Bytetrack: Multi-object tracking by associating every detection box. In: ECCV (2022)
54. Zhang, Y., Wang, C., Wang, X., Zeng, W., Liu, W.: Fairmot: On the fairness of detection and re-identification in multiple object tracking. IJCV (2021)
55. Zhang, Y., Wang, T., Zhang, X.: Motrv2: Bootstrapping end-to-end multi-object tracking by pretrained object detectors. In: CVPR (2023)
56. Zheng, S., Chen, S., Jin, Q.: Vrdformer: End-to-end video visual relation detection with transformers. In: CVPR (2022)
57. Zhou, L., Zhou, Y., Corso, J.J., Socher, R., Xiong, C.: End-to-end dense video captioning with masked transformer. In: CVPR (2018)
58. Zhou, X., Arnab, A., Sun, C., Schmid, C.: Dense video object captioning from disjoint supervision. arXiv (2023)
59. Zhou, X., Koltun, V., Krähenbühl, P.: Tracking objects as points. In: ECCV (2020)
60. Zhou, X., Yin, T., Koltun, V., Krähenbühl, P.: Global tracking transformers. In: CVPR (2022)
61. Zhu, P., Wen, L., Du, D., Bian, X., Fan, H., Hu, Q., Ling, H.: Detection and tracking meet drones challenge. TPAMI (2021)