Artificial Intelligence 293 (2021) 103448

Contents lists available at ScienceDirect

Artificial Intelligence

www.elsevier.com/locate/artint

Multiple object tracking: A literature review

Wenhan Luo [a,b], Junliang Xing [c,f,*], Anton Milan [d], Xiaoqin Zhang [e], Wei Liu [a], Tae-Kyun Kim [b]

[a] Tencent AI Lab, China
[b] Imperial College London, UK
[c] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China
[d] Amazon Research and Development Center, Germany
[e] Wenzhou University, China
[f] School of Artificial Intelligence, University of Chinese Academy of Sciences, China

ARTICLE INFO

Article history:
Received 28 February 2018
Received in revised form 11 December 2019
Accepted 23 December 2020
Available online 30 December 2020

Keywords:
Multi-object tracking
Data association
Survey

ABSTRACT

Multiple Object Tracking (MOT) has gained increasing attention due to its academic and commercial potential. Although different approaches have been proposed to tackle this problem, it still remains challenging due to factors like abrupt appearance changes and severe object occlusions. In this work, we contribute the first comprehensive and most recent review on this problem. We inspect the recent advances in various aspects and propose some interesting directions for future research. To the best of our knowledge, there has not been any extensive review on this topic in the community. We endeavor to provide a thorough review on the development of this problem in recent decades. The main contributions of this review are fourfold: 1) Key aspects in an MOT system, including formulation, categorization, key principles, evaluation of MOT are discussed; 2) Instead of enumerating individual works, we discuss existing approaches according to various aspects, in each of which methods are divided into different groups and each group is discussed in detail for the principles, advances and drawbacks; 3) We examine experiments of existing publications and summarize results on popular datasets to provide quantitative and comprehensive comparisons. By analyzing the results from different perspectives, we have verified some basic agreements in the field; and 4) We provide a discussion about issues of MOT research, as well as some interesting directions which will become potential research effort in the future.

1. Introduction

Multiple Object Tracking (MOT), or Multiple Target Tracking (MTT), plays an important role in computer vision. The task of MOT is largely partitioned into locating multiple objects, maintaining their identities, and yielding their individual trajectories given an input video. Objects to track can be, for example, pedestrians on the street [1,2], vehicles in the road [3,4], sport players on the court [5–7], or groups of animals (birds [8], bats [9], ants [10], fish [11–13], cells [14,15], bees [16], *etc.*). Multiple "objects" could also be viewed as different parts of a single object [17]. In this review, we mainly focus

* Corresponding author.
E-mail addresses: whluo.china@gmail.com (W. Luo), jlxing@nlpr.ia.ac.cn (J. Xing), antmila@amazon.com (A. Milan), zhangxiaoqinnan@gmail.com (X. Zhang), wl2223@columbia.edu (W. Liu), kimtaekyun@kaist.ac.kr (T.-K. Kim).

https://doi.org/10.1016/j.artint.2020.103448
0004-3702/

Table 1
A summary of other literature reviews.

Reference	Topic	Year
Zhan et al. [24]	Crowd Analysis	2008
Hu et al. [20]	Object Motion and Behaviors	2004
Kim et al. [25]	Intelligent Visual Surveillance	2010
Candamo et al. [22]	Behavior Recognition in Transit Scenes	2010
Xiaogang Wang [21]	Multi-Camera Video Surveillance	2013
Forsyth et al. [26]	Human Motion Analysis	2006
Kevin Cannons [27]	Visual Tracking	2008
Yilmaz et al. [28]	Object Visual Tracking	2006
Li et al. [29]	Appearance Models in Object Tracking	2013
Wu et al. [30]	Visual Tracking Benchmark	2013
Leal-Taixé et al. [31]	MOT Benchmark	2015
Zhao et al. [32]	Object Detection	2019

on the research on pedestrian tracking. The underlying reasons for this specification are threefold. First, compared to other common objects in our environment, pedestrians are typical non-rigid objects, which is an ideal example to study the MOT problem. Second, videos of pedestrians arise in a huge number of practical applications, which further results in great commercial potential. Third, according to all data collected for this review, at least 70% of current MOT research efforts are devoted to pedestrians.

As a mid-level task in computer vision, multiple object tracking grounds high-level tasks such as pose estimation [18], action recognition [19], and behavior analysis [20]. It has numerous practical applications, such as visual surveillance [21], human computer interaction [22] and virtual reality [23]. These practical requirements have sparked enormous interest in this topic. Compared with Single Object Tracking (SOT), which primarily focuses on designing sophisticated appearance models and/or motion models to deal with challenging factors such as scale changes, out-of-plane rotations and illumination variations, multiple object tracking additionally requires two tasks to be solved: determining the number of objects, which typically varies over time, and maintaining their identities. Apart from the common challenges in both SOT and MOT, further key issues that complicate MOT include among others: 1) frequent occlusions, 2) initialization and termination of tracks, 3) similar appearance, and 4) interactions among multiple objects. In order to deal with all these issues, a wide range of solutions have been proposed in the past decades. These solutions concentrate on different aspects of an MOT system, making it difficult for MOT researchers, especially newcomers, to gain a comprehensive understanding of this problem. Therefore, in this work we provide a review to discuss the various aspects of the multiple object tracking problem.

1.1. Differences from other related reviews

To the best of our knowledge, there has not been any comprehensive literature review on the topic of multiple object tracking. However, there have been some other reviews related to multiple object tracking, which are listed in Table 1. We group these surveys into four sets and highlight the differences from ours as follows.

- The first set [24,20,25,22,21] discusses tracking as an individual part while this work specifically discusses various aspects of MOT. For example, object tracking is discussed as a step in the procedure of high-level tasks such as crowd modeling [24,20,25]. Similarly, in [22] and [21], object tracking is reviewed as a part of a system for behavior recognition [22] or video surveillance [21].
- The second set [26–29] is dedicated to general visual tracking techniques [26–28] or some special issues such as appearance models in visual tracking [29]. Their reviewing scope is wider than ours; ours on the contrary is more comprehensive and focused on multiple object tracking.
- The third set [30,31] introduces and discusses benchmarks on general visual tracking [30] and on specific multiple object tracking [31]. Their attention is laid on experimental studies rather than literature reviews.
- The fourth set [32] reviews the recent advances and development in object detection with the rising of deep learning. The topic is related to ours but different from ours. Object detection can provide observations for detection-based object tracking by locating the potential object locations in each frame, while MOT needs to associate these observations across multiple frames to form the object trajectories.

1.2. Contributions

We provide the first comprehensive review on the MOT problem to the computer vision community, which we believe is helpful to understand this problem, its main challenges, pitfalls, and the state of the art. The main contributions of this review are summarized as follows:

Table 2
Denotations employed in this review.

Symbol	Description	Symbol	Description	Symbol	Description	Symbol	Description
P	probability	$\mathbf{I}$	image	$\mathbf{p}$	position	x, y	position
S	similarity	$\mathbf{S}$	set of states	$\mathbf{v}$	velocity	u, v	speed
C	cost	$\mathbf{O}$	set of observations	$\mathbf{f}$	feature	w, α, λ	weight
N	frame number	$\mathbf{T}$	trajectory/tracklet	$\mathbf{c}$	color	t	time index
M	object number	$\mathbf{M}$	feature matrix	$\mathbf{o}$	observation	i, j, k	general index
G	graph	$\mathbf{\Sigma}$	covariance matrix	$\mathbf{s}$	state	σ	variance
V	vertex set	$\mathbf{L}$	Laplacian matrix	$\mathbf{a}$	acceleration	ϵ	noise
E	edge set	$\mathbf{Y}$	label set	$\mathbf{y}$	label	s	size
D	distance						
L	likelihood						
F	function						
Z	normalization factor						
$\mathcal{N}$	normal distribution						
$\mathcal{S}$	set						

- We derive a unified formulation of the MOT problem which consolidates most of the existing MOT methods (Section 2.1), and two different ways to categorize MOT methods (Section 2.2).
- We investigate different key components involved in an MOT system, each of which is further divided into different aspects and discussed in detail regarding its principles, advances, and drawbacks (Section 3).
- Experimental results on popular datasets regarding different approaches are presented, which makes future experimental comparison convenient. By investigating the provided results, some interesting observations and findings are revealed (Section 4).
- By summarizing the MOT review, we unveil existing issues of MOT research. Furthermore, open problems are discussed to identify potential future research directions (Section 5).

Note that this work is mainly dedicated to reviewing recent literature on the advances in multiple object tracking. As mentioned above, we also present experimental results on publicly available datasets excepted from existing publications to provide a quantitative view on the state-of-the-art MOT methods. For standardized benchmarking of multiple object tracking we kindly refer the readers to the recent work MOTChallenge by Leal-Taixé et al. [31].

1.3. Organization of this review

Our goal is to provide an overview of the major aspects in the MOT task. These aspects include the current state of research in MOT, all the detailed issues requiring consideration in building a system, and how to evaluate an MOT system. Section 2 describes the MOT problem, including its general formulation (Section 2.1) and typical ways for categorization (Section 2.2). Section 3 contributes to the most common components involved in modeling multi-object tracking, *i.e.*, appearance model (Section 3.1), motion model (Section 3.2), interaction model (Section 3.3), exclusion model (Section 3.4), occlusion handling (Section 3.5), and inference methods (Section 3.6). Furthermore, issues concerning evaluations, including metrics (Section 4.1), public datasets (Section 4.2), public codes (Section 4.3), and benchmark results (Section 4.4) are discussed in Section 4. This part is followed by Section 5 which summarizes the existing issues and interesting problems for future directions of the MOT research in the community.

1.4. Denotations

Throughout this manuscript, we denote scalar and vector variables by lowercase letters (*e.g.*, x) and bold lowercase letters (*e.g.*, $\mathbf{x}$), respectively. We use bold capital letters (*e.g.*, $\mathbf{X}$) to denote a matrix or a set of vectors. Capital letters (*e.g.*, X) are adopted for specific functions or variables. Table 2 lists symbols utilized throughout this review. Except the symbols in the table, there may be some symbols for a specific reference. As these symbols are not commonly employed, they are not listed in the table but will be rather defined in the context.

2. MOT problem

We first endeavor to give a general mathematical formulation of MOT. We then discuss its possible categorizations based on different aspects.

2.1. Problem formulation

The MOT problem has been formulated differently from various perspectives in previous works, which makes it difficult to understand this problem from a high-level view. Here we offer a general formulation and argue that existing works can be unified under this formulation. To the best of our knowledge, there has not been any previous work towards this attempt.

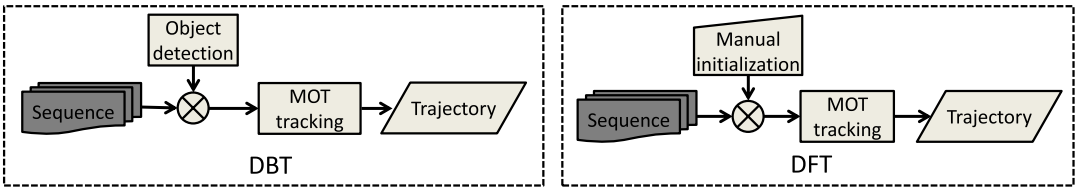

Fig. 1. A procedure flow of two prominent tracking approaches. **Left:** Detection-Based Tracking (DBT), **right:** Detection-Free Tracking (DFT).

In general, multiple object tracking can be viewed as a multi-variable estimation problem. Given an image sequence, we employ $\mathbf{s}_t^i$ to denote the state of the i-th object in the t-th frame, $\mathbf{S}_t = (\mathbf{s}_t^1, \mathbf{s}_t^2, ..., \mathbf{s}_t^{M_t})$ to denote states of all the M_t objects in the t-th frame. We employ $\mathbf{s}_{i_s:i_e}^i = \{\mathbf{s}_{i_s}^i, ..., \mathbf{s}_{i_e}^i\}$ to denote the sequential states of the i-th object, where i_s and i_e are respectively the first and last frame in which target i exists, and $\mathbf{S}_{1:t} = \{\mathbf{S}_1, \mathbf{S}_2, ..., \mathbf{S}_t\}$ to denote all the sequential states of all the objects from the first frame to the t-th frame. Note that the object number may vary from frame to frame.

Correspondingly, following the most commonly used tracking-by-detection, or Detection Based Tracking (DBT) paradigm, we utilize $\mathbf{o}_t^i$ to denote the collected observations for the i-th object in the t-th frame, $\mathbf{O}_t = (\mathbf{o}_t^1, \mathbf{o}_t^2, ..., \mathbf{o}_t^{M_t})$ to denote the collected observations for all the M_t objects in the t-th frame, and $\mathbf{O}_{1:t} = \{\mathbf{O}_1, \mathbf{O}_2, ..., \mathbf{O}_t\}$ to denote all the collected sequential observations of all the objects from the first frame to the t-th frame.

The objective of multiple object tracking is to find the "optimal" sequential states of all the objects, which can be generally modeled by performing MAP (Maximum a posteriori) estimation from the conditional distribution of the sequential states given all the observations:

$$\widehat{\mathbf{S}}_{1:t} = \arg\max_{\mathbf{S}_{1:t}} P(\mathbf{S}_{1:t}|\mathbf{O}_{1:t}). \tag{1}$$

Different MOT algorithms from previous works can now be thought as designing different approaches to solving the above MAP problem, either from a *probabilistic inference* perspective [33,6,34–39] or a *deterministic optimization* perspective [40–50,17].

The probabilistic inference based approaches usually solve the MAP problem in Eq. (1) using a two-step iterative procedure as follows,

Predict: $P(\mathbf{S}_t|\mathbf{O}_{1:t-1}) = \int P(\mathbf{S}_t|\mathbf{S}_{t-1}) P(\mathbf{S}_{t-1}|\mathbf{O}_{1:t-1}) d\mathbf{S}_{t-1}$,

Update: $P(\mathbf{S}_t|\mathbf{O}_{1:t}) \propto P(\mathbf{O}_t|\mathbf{S}_t) P(\mathbf{S}_t|\mathbf{O}_{1:t-1})$.

Here $P(\mathbf{S}_t|\mathbf{S}_{t-1})$ and $P(\mathbf{O}_t|\mathbf{S}_t)$ are the *Dynamic Model* and the *Observation Model*, respectively.

The deterministic optimization based approaches directly maximize the likelihood function $L(\mathbf{O}_{1:t}|\mathbf{S}_{1:t})$ as a delegate of $P(\mathbf{S}_{1:t}|\mathbf{O}_{1:t})$ over a set of available observations $\{\hat{\mathbf{O}}_{1:t}^n\}$:

$$\widehat{\mathbf{S}}_{1:t} = \arg\max_{\mathbf{S}_{1:t}} P(\mathbf{S}_{1:t}|\mathbf{O}_{1:t}) = \arg\max_{\mathbf{S}_{1:t}} L(\mathbf{O}_{1:t}|\mathbf{S}_{1:t}) = \arg\max_{\mathbf{S}_{1:t}} \prod_n P\left(\hat{\mathbf{O}}_{1:t}^n|\mathbf{S}_{1:t}\right), \tag{2}$$

or conversely minimize an energy function $E(\mathbf{S}_{1:t}|\mathbf{O}_{1:t})$:

$$\widehat{\mathbf{S}}_{1:t} = \arg\max_{\mathbf{S}_{1:t}} P(\mathbf{S}_{1:t}|\mathbf{O}_{1:t}) = \arg\max_{\mathbf{S}_{1:t}} \exp\left(-E(\mathbf{S}_{1:t}|\mathbf{O}_{1:t})\right)/Z = \arg\min_{\mathbf{S}_{1:t}} E(\mathbf{S}_{1:t}|\mathbf{O}_{1:t}), \tag{3}$$

where Z is a normalization factor to ensure $P(\mathbf{S}_{1:t}|\mathbf{O}_{1:t})$ to be a probability distribution.

2.2. MOT categorization

It is difficult to classify one particular MOT method into a distinct category by a universal criterion. Admitting this, it is thus feasible to group MOT methods by multiple criteria. In the following we attempt to conduct this according to three criteria: *a) initialization method*, *b) processing mode*, and *c) type of output*. The reason we choose these three criteria is that this naturally follows the way of processing a task, *i.e.*, how the task is initialized, how it is processed and what type of result is obtained. We believe that other criteria may also be reasonably adopted to categorize various MOT methods. However, categorizing different MOT methods with all possible criteria are beyond the scope of this article. In the following, each of the above criteria along with its corresponding categorization is represented.

2.2.1. Initialization method

Most existing MOT works can be grouped into two sets [51], depending on how objects are initialized: Detection-Based Tracking (DBT) and Detection-Free Tracking (DFT).

Detection-based tracking. As shown in Fig. 1 (top), objects are first detected and then linked into trajectories. This strategy is also commonly referred to as "tracking-by-detection". Given a sequence, type-specific object detection or motion detection (based on background modeling) [52,53] is applied in each frame to obtain object hypotheses, then (sequential or batch) tracking is conducted to link detection hypotheses into trajectories. There are two issues worth noting. First, since the

Table 3
Comparison between DBT and DFT. Adapted from [51].

Item	DBT	DFT
Initialization	automatic, imperfect	manual, perfect
# of objects	varying	fixed
Applications	specific type of objects (in most cases)	any type of objects
Advantages	ability to handle varying number of objects	free of object detector
Drawbacks	performance depends on object detection	manual initialization

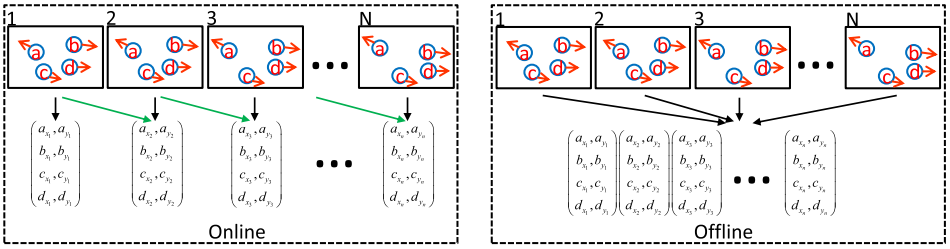

Fig. 2. An illustration of online (left) and offline (right) tracking. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

Table 4
Comparison between online and offline tracking.

Item	Online tracking	Offline tracking
Input	Up-to-time observations	All observations
Methodology	Gradually extend existing trajectories with current observations	Link observations into trajectories
Advantages	Suitable for online tasks	Obtain global optimal solution theoretically
Drawbacks	Suffer from shortage of observation	Delay in outputting final results

object detector is trained in advance, the majority of DBT focuses on specific kinds of targets, such as pedestrians, vehicles or faces. Second, the performance of DBT highly depends on the performance of the employed object detector.

Detection-free tracking. As shown in Fig. 1 (bottom), DFT [54–57] requires manual initialization of a fixed number of objects in the first frame, then localizes these objects in subsequent frames.

DBT is more popular because new objects are discovered and disappearing objects are terminated automatically. DFT cannot deal with the case that objects appear. However, it is free of pre-trained object detectors. Table 3 lists the major differences between DBT and DFT.

2.2.2. Processing mode

MOT can also be categorized into *online* tracking and *offline* tracking. The difference is whether observations from future frames are utilized when handling the current frame. Online, also called causal, tracking methods only rely on the past information available up to the current frame, while offline, or batch tracking approaches employ observations both in the past and in the future.

Online tracking. In online tracking [54,58,55,56,59,60,165,160], the image sequence is handled in a step-wise manner, thus online tracking is also named as sequential tracking. An illustration is shown in Fig. 2 (top), with three objects (different circles) a, b, and c. The green arrows represent observations in the past. The results are represented by the object's location and its ID. Based on the up-to-time observations, trajectories are produced on the fly.

Offline tracking. Offline tracking [53,61,49,62,48,1,63–66] utilizes a batch of frames to process the data. As shown in Fig. 2 (bottom), observations from all the frames are required to be obtained in advance and are analyzed jointly to estimate the final output. Note that, due to computational and memory limitation, it is not always possible to handle all the frames at once. An alternative solution is to split the data into shorter video clips, and infer the results hierarchically or sequentially for each batch. Table 4 lists the differences between the two processing modes.

2.2.3. Type of output

This criterion classifies MOT methods into deterministic ones and probabilistic ones, depending on the randomness of output. The difference between these two types of methods primarily results from the optimization methods adopted as mentioned in Section 2.1.

Stochastic tracking. The output results of stochastic tracking vary from time to time. For example, in the case of detection-free tracking, the bounding box results are different if we utilize particle filter for inference. The difference results from the randomness of the generation of particles in the processing. Even in the case of detection-based tracking, some

studies also employ state-of-the-art single object tracker to refine the detection bounding boxes. This kind of methods will also lead to different tracking results in different running times.

Deterministic tracking. The output of deterministic tracking is constant when running the methods multiple times. For instance, in the case of tracking-by-detection, data association methods like Hungarian algorithm will produce deterministic tracking results. Deterministic tracking usually is associated with deterministic optimization for deriving the final output.

2.2.4. Discussion

The difference between DBT and DFT is whether a detection model is adopted (DBT) or not (DFT). The key to differentiate online and offline tracking is the way they process observations. Readers may question whether DFT is identical to online tracking because it seems DFT always processes observations sequentially. This is true in most cases although some exceptions exist. Orderless tracking [67] is an example. It is DFT and simultaneously processes observations in an orderless way. Though it is for single object tracking, it can also be applied for MOT, and thus DFT can also be applied in a batch mode. Another vagueness may rise between DBT and offline tracking, as in DBT tracklets or detection responses are usually associated in a batch way. Note that there are also sequential DBT which conducts association between previously obtained trajectories and new detection responses [8,68,33].

The categories presented above in Section 2.2.1, 2.2.2 and 2.2.3 are three possible ways to classify MOT methods, while there may be others. Notably, specific solutions for sport scenarios [6,5], aerial scenes [69,46], generic objects [68,70,71,8,72], *etc*. exist, and we suggest the readers refer to the respective publications.

By providing these three criteria described above, it is convenient for one to tag a specific method with the combination of the categorization label. This would help one to understand a specific approach easier.

3. MOT components

In this section, we represent the primary components of an MOT approach. As mentioned above, the goal of MOT is to discover multiple objects in individual frames and recover the identity information across continuous frames, *i.e*., trajectory, from a given sequence. When developing MOT approaches, two major issues should be considered. One is how to measure similarity between objects in frames, the other one is how to recover the identity information based on the similarity measurement between objects across frames. Roughly speaking, the first issue involves the modeling of appearance, motion, interaction, exclusion, and occlusion. The second one involves with the inference problem. We review recent progress regarding both items in the following.

3.1. Appearance model

Appearance is an important cue for affinity computation in MOT. However, different from single object tracking, which primarily focuses on constructing a sophisticated appearance model to discriminate object from background, most MOT methods do not consider appearance modeling as the core component, although it can be an important one.

Technically, an appearance model includes two components: *visual representation* and *statistical measuring*. Visual representation describes the visual characteristics of an object using some features, either based on a single cue or multiple cues. Statistical measuring, on the other hand, is the computation of similarity between different observations. More formally, the similarity between two observations i and j can be written as

$$S_{ij} = F\left(\mathbf{o}_i, \mathbf{o}_j\right), \tag{4}$$

where $\mathbf{o}_i$ and $\mathbf{o}_j$ are visual representations of different observations, and $F(\cdot,\cdot)$ is a function that measures the similarity between them. In the following, we first discuss the visual representation in MOT, and then describe statistical measurement, respectively.

3.1.1. Visual representation

Visual representation describes an object according to different kinds of features, which are shown in Fig. 3. We group features into the following different categories.

Local features. KLT is an example of searching "good" local features and tracking. It is successfully adopted in both SOT [77] and MOT. Obtaining easy-to-track features, we can employ them to generate short trajectories [65,78], estimate camera motion [66,79], motion clustering [71] and so on. Optical flow can also be regarded as local features if we treat image pixel as the finest local range. A set of solutions to MOT utilize optical flow to link detection responses into short tracklets before data association [80,81]. As it is related with motion, it is utilized to encode motion information [82,83]. One special application of optical flow is to discover crowd motion patterns in packed scenarios [73,37], where ordinary features are not reliable.

Region features. Compared with local features, region features are extracted from a wider range (*e.g*. a bounding box). We illustrate them as three types: a) *zero-order* type, b) *first-order* type and c) *up-to-second-order* type. Here, order means the order of discrepancy when computing the representation. For instance, zero-order means values of pixels are not compared, while one-order means discrepancy values among pixels are computed once.

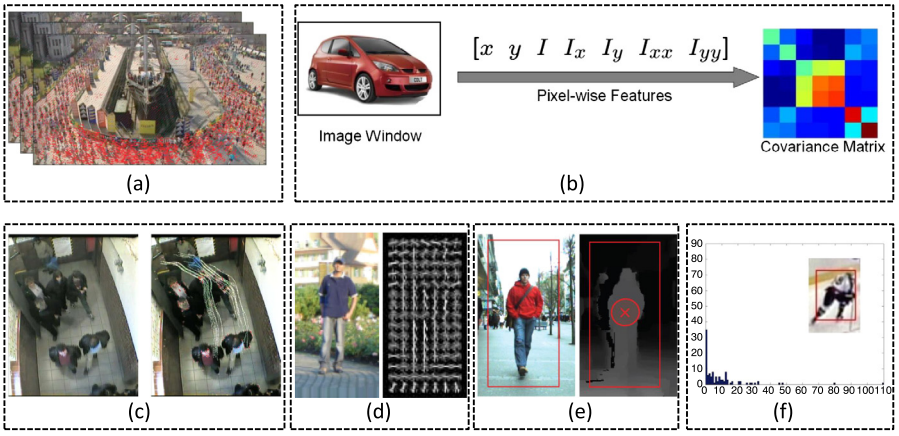

Fig. 3. An illustration of various visual features. (a) Optical flow [73], (b) covariance matrix, (c) point features [71], (d) gradient based features (HOG) [74], (e) depth [75], and (f) color features [76]. Best viewed in color.

- *Zero-order.* This is the most widely utilized representation for MOT. Color histogram [65,75,81,76,36,84] and raw pixel template [85] are two typical examples of this type.
- *First-order.* Gradient-based representations like HOG [81,63,34,19,86] and level-set formulation [75] are commonly employed.
- *Up-to-second-order.* Region covariance matrix [87,88] which computes up-to-second-order discrepancy belongs to this set. It has been adopted in [64,63,54].

Others. Besides local and region features, there are some other kinds of representation. Taking depth as an example, it is typically used to refine detection hypotheses [75,89–92]. The Probabilistic Occupancy Map (POM) [93,44] is employed to estimate how likely an object would occur in a specific grid cell. One more example is gait feature, which is unique for individual persons [65]. DCNN [94] plays a role of a codebook similar to bag-of-words (BoW) in [95]. ColorNames descriptor is utilized in [96] for appearance representation. Deep features from Convolution Neural Network (CNN) are employed for visual representation in [97,98]. In [99], point cloud feature is the first time to be introduced for feature fusion in MOT.

Discussion. Generally, color histogram is a well studied similarity measure, but it ignores the spatial layout of the object region. Local features are efficient, but sensitive to issues like occlusion and out-of-plane rotation. Gradient based features like HOG can describe the shape of an object and are robust to certain transformations such as illumination changes, but they cannot handle occlusion and deformation well. Region covariance matrix features are more robust as they take more information in account, but this benefit is obtained at the cost of more computation. Depth features make the computation of affinity more accurate, but they require multiple views of the same scenery and/or additional algorithm [100] to obtain depth measurements.

3.1.2. Statistical measuring

This step is closely related to the section above. Based on visual representation, statistical measure computes the affinity between two observations. While some approaches solely rely on one kind of cue, others are built on multiple cues.

Single cue. Modeling appearance using single cue is either transforming distance into similarity or directly calculating the affinity. For example, the Normalized Cross Correlation (NCC) is usually adopted to calculate the affinity between two counterparts based on the representation of raw pixel template mentioned above [85,73,101,2]. Speaking of color histogram, Bhattacharyya distance $B(\cdot,\cdot)$ is used to compute the distance between two color histograms $\mathbf{c}_i$ and $\mathbf{c}_j$. The distance is transformed into similarity S like $S(\mathbf{T}_i, \mathbf{T}_j) = \exp(-B(\mathbf{c}_i, \mathbf{c}_j))$ [38,65,66,102,61,33] or fit the distance to Gaussian distributions like [40]. Transformation of dissimilarity into likelihood is also applied to the representation of covariance matrix [64]. Cosine similarity between deep features from neural network is used in [103]. Besides these typical models, bag-of-words model [104] is employed based on point feature representation [35].

Multiple cues. Different kinds of cues can complement each other to make the appearance model more robust. However, it not trivial to decide how to fuse the information from multiple cues. Regarding this, we summarize multi-cue based appearance models according to five kinds of fusion strategies: *Boosting*, *Concatenating*, *Summation*, *Product*, and *Cascading* (see also Table 5).

- Boosting. The strategy of Boosting usually selects a portion of features from a feature pool sequentially via a Boosting based algorithm. For example, from color histogram, HOG and covariance matrix descriptor, AdaBoost, RealBoost, and a HybridBoost algorithm are respectively employed to choose the most representative features to discriminate pairs of tracklets of the same object from those of different objects in [63], [51] and [42].
- Concatenation. Different kinds of features can be concatenated for computation. In [48], color, HOG and optical flow are concatenated for appearance modeling.

Table 5
An overview of typical appearance models employing multiple cues.

Strategy	Employed Cue	Ref.
Boosting	Color, HOG, shapes, covariance matrix, *etc.*	[63,42,51]
Concatenating	Color, HOG, optical flow, *etc.*	[48]
Summation	Color, depth, correlogram, LBP, *etc.*	[75,105,106]
Product	Color, shapes, bags of local features, *etc.*	[35,53,107,108]
Cascading	Depth, shape, texture, *etc.*	[92,81]

- Summation. This strategy takes affinity values from different features and balances these values with weights [75,105, 106].
- Product. Differently from the strategy above, values are multiplied to produce the integrated affinity [35,53,107,108]. Note that, independence assumption is usually made when applying this strategy.
- Cascading. This is a cascade manner of using various types of visual representation, either to narrow the search space [92] or model appearance in a coarse-to-fine way [81].

3.2. Motion model

The motion model captures the dynamic behavior of an object. It estimates the potential position of objects in the future frames, thereby reducing the search space. In most cases, objects are assumed to move smoothly in the world and therefore in the image space (except for abrupt motions). We will discuss linear motion model and non-linear motion model in the following.

3.2.1. Linear motion model

This is by far the most popular model [109,110,34]. A constant velocity assumption [34] is made in this model. Based on this assumption, there are three different ways to construct the model.

- Velocity smoothness is modeled by enforcing the velocity values of an object in successive frames to change smoothly. In [47], it is implemented as a cost term,

$$C_{dyn} = \sum_{t=1}^{N-2} \sum_{i=1}^{M} \left\| \mathbf{v}_i^t - \mathbf{v}_i^{t+1} \right\|^2, \tag{5}$$

  where the summation is conducted over N frames and M trajectories/objects.
- Position smoothness directly forces the discrepancy between the observed position and estimated position. Let us take [33] as an example. Considering a temporal gap Δt between tail of tracklet $\mathbf{T}_i$ and head of tracklet $\mathbf{T}_j$, the smoothness is modeled by fitting the estimated position to a Gaussian distribution with the observed position as center. In the stage of estimation, both forward motion and backward motion are considered. Thus, the affinity considering linear motion model is,

$$P_m\left(\mathbf{T}_i, \mathbf{T}_j\right) = \mathcal{N}\left(\mathbf{p}_i^{tail} + \mathbf{v}_i^F \Delta t; \mathbf{p}_j^{head}, \Sigma_j^B\right) * \mathcal{N}\left(\mathbf{p}_j^{head} + \mathbf{v}_i^B \Delta t; \mathbf{p}_i^{tail}, \Sigma_i^F\right), \tag{6}$$

  where "F" and "B" means forward and backward direction. A similar strategy is adopted by Yang et al. [62]. The displacement between observed position and estimated position $\Delta\mathbf{p}$ is fit to a Gaussian distribution with zero center. Other examples of this strategy are [63,111,1,61,7,62].
- Acceleration smoothness. Besides considering position and velocity smoothness, acceleration is taken into account [111]. The probability distribution of motion of a state $\{\hat{\mathbf{s}}_k\}$ at time k given the observation tracklet $\{\mathbf{o}_k\}$ is modeled as,

$$P\left(\{\hat{\mathbf{s}}_k\} \mid \{\mathbf{o}_k\}\right) = \prod_k \mathcal{N}\left(\mathbf{x}_k - \hat{\mathbf{x}}_k; \mathbf{0}, \Sigma_p\right) \prod_k \mathcal{N}\left(\mathbf{v}_k; \mathbf{0}, \Sigma_{\mathbf{v}}\right) \prod_k \mathcal{N}\left(\mathbf{a}_k; \mathbf{0}, \Sigma_{\mathbf{a}}\right), \tag{7}$$

  where $\mathbf{v}_k$ is the velocity, $\mathbf{a}_k$ is the acceleration, and $\mathcal{N}$ is a zero-mean Gaussian distribution.

3.2.2. Non-linear motion model

The linear motion model is commonly used to explain the object's dynamics. However, there are some cases which the linear motion model cannot deal with. To this end, non-linear motion models are proposed to produce more accurate motion affinity between tracklets. For instance, Yang et al. [49] employ a non-linear motion model to handle the situation that targets may move freely. Given two tracklets $\mathbf{T}_1$ and $\mathbf{T}_2$ which belong to the same target in Fig. 4(a), the linear motion model [62] would produce a low probability to link them. Alternatively, employing the non-linear motion model, the gap between the tail of tracklet $\mathbf{T}_1$ and the head of tracklet $\mathbf{T}_2$ could be reasonably explained by a tracklet $\mathbf{T}_0 \in \mathcal{S}$, where $\mathcal{S}$ is the set of support tracklets. As shown in Fig. 4(b), $\mathbf{T}_0$ matches the tail of $\mathbf{T}_1$ and the head of $\mathbf{T}_2$. Then the real path to bridge $\mathbf{T}_1$ and $\mathbf{T}_2$ is estimated based on $\mathbf{T}_0$, and the affinity between $\mathbf{T}_1$ and $\mathbf{T}_2$ is computed similarly as described in Section 3.2.1.

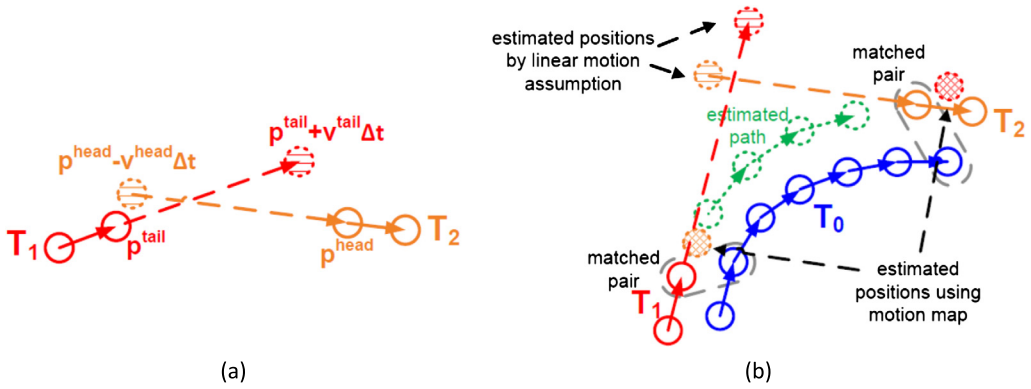

Fig. 4. An image comparing the linear motion model (a) with the non-linear motion model (b) [49]. Best viewed in color.

Table 6
Existing publications employing social force models.

Ref.	Employed Forces
[2]	repulsion, constancy, fidelity
[85]	repulsion, constancy, attraction, coherence
[61]	coherence
[66]	repulsion, coherence
[115]	constancy, fidelity, repulsion
[116]	constancy, coherence

3.3. Interaction model

Interaction model, also known as mutual motion model, captures the influence of an object on other objects. In the crowd scenery [84], an object would experience some "force" from other agents and objects. For instance, when a pedestrian is walking on the street, he would adjust his speed, direction and destination, in order to avoid collisions with others. Another example is when a crowd of people walks across a street, each of them follows other people and guides others at the same time. In fact, these are examples of two typical interaction models known as the *social force models* [112] and the *crowd motion pattern models* [113].

3.3.1. Social force models

Social force models are also known as group models. In these models, each object is considered to be dependent on other objects and environmental factors. This type of information could alleviate performance deterioration in crowded scenes. In social force models, targets are considered as agents which determine their velocity, acceleration, and destination based on observations of other objects and the environment. More specifically, in social force models, target behavior is modeled based on two aspects, *individual force* and *group force*.

Individual force. For each individual in a group of multiple objects, two types of forces are considered:

- ***fidelity***, which means one should not change his desired destination
- ***constancy***, which means one should not suddenly change his momentum, including speed and direction

Group force. For a whole group, three types of forces are considered:

- ***attraction***, which means individuals moving together as a group should stay close
- ***repulsion***, which means that individuals moving together as a group should keep some distance away from others to make all members comfortable
- ***coherence***, which means individuals moving together as a group should move with similar velocity

The majority of existing publications modeling interaction among objects with social force typically minimizes an energy objective, consisting of terms reflecting individual force and group force. Table 6 lists exemplar publications in the community which adopt social force models for interaction modeling. While [114] is an exception of explicitly modeling social force as energy terms. The social force is encoded as the so-called social feature for further processing in this study.

3.3.2. Crowd motion pattern models

Inspired by the crowd simulation literature [24], motion patterns are introduced to alleviate the difficulty of tracking an individual object in the crowd. In general, this type of models is usually applied in the over-crowded scenario where the density of targets is considerably high. In such highly-crowded scenery, objects are usually quite small, and cues such as appearance and individual motion are ambiguous. In this case, motion from the crowd is a comparably reliable cue for the problem.

Roughly, there are two kinds of motion patterns, *structured* and *unstructured* ones. Structured motion patterns exhibit collective spatio-temporal structure while unstructured motion patterns exhibit various modalities of motion. In general, motion patterns are learned by various methods (including ND tensor voting [78], Hidden Markov Models [38,117], Correlated Topic Model [80], sometimes considering scene structures [73]) and applied as prior knowledge to assist object tracking.

3.4. Exclusion model

Exclusion is a constraint employed to avoid physical collisions when seeking a solution to the MOT problem. It arises from the fact that two distinct objects cannot occupy the same physical space in the real world. Given multiple detection responses and multiple trajectory hypotheses, generally there are two constraints. The first one is the so-called *detection-level exclusion* [118], *i.e.*, two different detection responses in the same frame cannot be assigned to the same target. The second one is the so-called *trajectory-level exclusion*, *i.e.*, two trajectories cannot be infinitely close to each other.

3.4.1. Detection-level exclusion modeling

Different approaches are adopted to model the detection-level exclusion. Basically, there are "soft" and "hard" models.

"Soft" modeling. Detection-level exclusion is "softly" modeled by minimizing a cost term to penalize the case of violation. For example, a penalty is defined if two simultaneous detection responses are assigned the same label of trajectory and they are sufficiently distant from each other in [118].

To model exclusion, a special exclusion graph is constructed to capture the constraint [119]. Given all the detection responses, they define a graph where nodes represent detection responses. Each node (one detection) is connected only to nodes (other detections) that exist at the same time as the node itself. After constructing this graph, the label assignment is maximized *w.r.t.* exclusion to encourage connected nodes to have different labels as $\mathrm{Tr}(\mathbf{YLY})$, where $\mathbf{L}$ is the Laplacian matrix, $\mathbf{Y} = \left(\mathbf{y}_1, ..., \mathbf{y}_{|V|}\right)$ is the label assignment of all the $|V|$ nodes in the graph, and $\mathrm{Tr}(\cdot)$ is the trace norm of a matrix.

"Hard" modeling. "Hard" modeling of detection-level exclusion is implemented by applying explicit constraint. For instance, to model the detection-level exclusion, the so-called cannot links are introduced to imitate that if two tracklets have overlap in their time span, then they cannot be assigned to the same cluster, *i.e.* to belong to the same trajectory [120]. Non-negative discretization is conducted in [121] to set detections into non-overlapping groups to obey the constraint of mutual exclusion.

3.4.2. Trajectory-level exclusion modeling

Generally, trajectory-level exclusion is modeled by penalizing the case that two close detection hypotheses have different trajectory labels. This will suppress one trajectory label. For example, the penalty term in [122] is inversely proportional to the distance between two detection responses with different trajectory labels. If two detection responses are too close, it will lead to a considerably large, or in the limit case, infinite, cost. A similar idea is adopted in [50]. The penalty of trajectory-level exclusion in [118] is proportional to the spatial-temporal overlap between two trajectories. The closer the two trajectories, the higher the penalty. There is also a special case [45], in which the exclusion is modeled as an extra constraint to the so-called "Conflict" edges in a network flow based algorithm.

3.5. Occlusion handling

Occlusion is perhaps the most critical challenge in MOT. It is a primary cause for ID switches or fragmentation of trajectories. In order to handle occlusion, various kinds of strategies have been proposed.

3.5.1. Part-to-whole

This strategy is built on the assumption that a part of the object is still visible when an occlusion happens. This assumption holds in most cases. Based on this assumption, approaches adopting this strategy observe and utilize the visible part to infer the state of the whole object.

The popular way is dividing a holistic object (like a bounding box) into several parts and computing affinity based on individual parts. If an occlusion happens, affinities regarding occluded parts should be low. Tracker would be aware of this and adopt only the unoccluded parts for estimation. Specifically, parts are derived by dividing objects into grids uniformly [54], or fitting multiple parts into a specific kind of object like human, *e.g.* 15 non-overlap parts as in [51], and parts detected from the DPM detector [123] in [81,124].

Based on these individual parts, observations of the occluded parts are ignored. For instance, part-wise appearance model is constructed in [54]. Reconstructed error is used to determine which part is occluded or not. The appearance model of the holistic object is selectively updated by only updating the unoccluded parts. This is the "hard" way of ignoring the occluded part, while there is a "soft" way in [51]. Specifically, the affinity concerning two tracklets j and k is computed as $\sum_i w_i F\left(\mathbf{f}_j^i, \mathbf{f}_k^i\right)$, where $\mathbf{f}$ is feature, i is the index of parts. The weights are learned according to the occlusion relationship of parts. In [81], human body part association is conducted to recover the part trajectory and further assists whole object trajectory recovery.

Fig. 5. Training samples for the double-person detector [126]. From left to right, the level of occlusion increases.

"Part-to-whole" strategy is also applied in tracking based on feature point clustering, which assumes feature points with similar motion should belong to the same object. As long as some parts of an object are visible, the clustering of feature point trajectories will work [65,71,125].

3.5.2. Hypothesize-and-test

This strategy sidesteps challenges from occlusion by hypothesizing proposals and testing the proposals according to observations at hand. As the name indicates, this strategy is composed of two steps, *hypothesize* and *test*.

Hypothesize. Zhang et al. [40] generate occlusion hypotheses based on the occludable pair of observations, which are close and with similar scale. Assuming $\mathbf{o}_i$ is occluded by $\mathbf{o}_j$, a corresponding occlusion hypothesis is $\widetilde{\mathbf{o}}_i^j = (\mathbf{p}_j, s_i, \mathbf{f}_i, t_j)$, where $\mathbf{p}_j$ and t_j are the position and time stamp of $\mathbf{o}_j$, and s_i and $\mathbf{f}_i$ are the size and appearance feature of $\mathbf{o}_i$. This approach treats occlusion as distraction, while in other works [127,126] occlusion patterns are employed to assist detection in case of occlusion. More specifically, different detection hypotheses are generated by synthetically combining two objects with different levels and patterns of occlusion (see Fig. 5).

Test. The hypotheses would be employed for MOT when they are ready. Let us revisit the two approaches described above. In [40], the hypothesized observations along with the original ones are given as input to the cost-flow framework and MAP is conducted to obtain the optimal solution. In [127] and [126], a multi-person detector is trained based on the detection hypotheses. This detector greatly reduces the difficulty of detection in case of occlusion.

3.5.3. Buffer-and-recover

This strategy buffers observations when occlusion happens and remembers states of objects before occlusion. When occlusion ends, object states are recovered based on the buffered observations and the stored states before occlusion.

Mitzel et al. [75] keep a trajectory alive for up to 15 frames when occlusion happens, and extrapolates the position to grow the dormant trajectory through occlusion. In case the object reappears, the track is triggered again and the identity is maintained. This idea is followed in [36]. Observation mode is activated when the tracking state becomes ambiguous due to occlusion [128]. As soon as enough observations are obtained, hypotheses are generated to explain the observations. This could also be treated as "buffer-and-recover" strategy.

3.5.4. Others

The strategies described above may not cover all the tactics explored in the community. For example, Andriyenko et al. [129] represent targets as Gaussian distributions in image space and explicitly model pair-wise occlusion ratios between all targets as part of a differentiable energy function. In general, it is non-trivial to distinctly separate or categorize various approaches to occlusion modeling, and in some cases, multiple strategies are used in combination.

3.6. Inference

3.6.1. Probabilistic inference

Approaches based on probabilistic inference [161,164] typically represent states of objects as a distribution with uncertainty. The goal of a tracking algorithm is to estimate the probabilistic distribution of target state by a variety of probability reasoning methods based on existing observations. This kind of approaches typically requires only the existing, *i.e.* past and present observations, thus they are especially appropriate for the task of online tracking. As only the existing observations are employed for estimation, it is natural to impose the assumption of Markov property in the objects state sequence. This assumption includes two aspects, recalling the formula in Section 2.1.

First, the current object state only depends on the previous states. Further, it only depends on the very last state if the first-order Markov property is imposed, which can be formalized as $P(\mathbf{S}_t|\mathbf{S}_{1:t-1}) = P(\mathbf{S}_t|\mathbf{S}_{t-1})$.

Second, the observation of an object is only related to its state corresponding to this observation. In other words, the observations are conditionally independent: $P(\mathbf{O}_{1:t}|\mathbf{S}_{1:t}) = \prod_{i=1}^{t} P(\mathbf{O}_i|\mathbf{S}_i)$.

These two aspects are related to the *Dynamic Model* and the *Observation Model*, respectively. The dynamic model corresponds to the tracking strategy, while the observation model provides observation measurements concerning object states. The *predict* step is to estimate the current state based on all the previous observations. More specifically, the posterior

probability distribution of the current state is estimated by integrating in the space of the last object state via the dynamic model. The *update* step is to update the posterior probability distribution of states based on the obtained measurements under the observation model.

According to the equations, states of objects can be estimated by iteratively conducting the prediction and updating steps. However, in practice, the object state distribution cannot be represented without simplifying assumptions, thus there is no analytical solution to computing the integral of the state distribution. Additionally, for multiple objects, the dimension of the sets of states is very large, which makes the integration even more difficult, requiring the derivation for approximate solutions.

Various kinds of probabilistic inference models haves been applied to multi-object tracking [38,130,107,131], such as Kalman filter [37,39], Extended Kalman filter [36] and Particle filter [132–135,54,105,34,35].

Kalman filter. In the case of a linear system and Gaussian-distributed object states, the Kalman filter [39] is proven to be the optimal estimator. It has been applied in [37].

Extended Kalman filter. To include the non-linear case, the extended Kalman filter is one possible solution. It approximates the non-linear system by a Taylor expansion [36].

Particle filter. Monte Carlo sampling based models have also become popular in tracking, especially after the introduction of the particle filter [132–134,54,105,34,35,10]. This strategy models the underlying distribution by a set of weighted particles, thereby allowing to drop any assumptions about the distribution itself [105,34,35,38].

3.6.2. Deterministic optimization

As opposed to the probabilistic inference methods, approaches based on deterministic optimization aim to find the maximum a posteriori (MAP) solution to MOT. To that end, the task of inferring data association, the target states or both, is typically cast as an optimization problem. Approaches within this framework are more suitable for the task of offline tracking because observations from all the frames or at least a time window are required to be available in advance. Given observations (usually detection hypotheses) from all the frames, these types of methods endeavor to globally associate observations belonging to an identical object into a trajectory. The key issue is how to find the optimal association. Some popular and well-studied approaches are detailed in the following.

Bipartite graph matching. By modeling the MOT problem as bipartite graph matching, two disjoint sets of graph nodes could be existing trajectories and new detections in online tracking or two sets of tracklets in offline tracking. Weights among nodes are modeled as affinities between trajectories and detections. Then either a greedy bipartite assignment algorithm [124,34,136] or the optimal Hungarian algorithm [61,69,137,33,41] are employed to determine the matching between nodes in the two sets.

Dynamic programming. Extend dynamic programming [138], linear programming [139–141], quadratic boolean programming [142], K-shortest paths [44,19], set cover [143] and subgraph multicut [144–146], maximum multi-clique [147] are adopted to solve the association problem among detections or tracklets.

Min-cost max-flow network flow. Network flow is a directed graph where each edge has a certain capacity. For MOT, nodes in the graph are detection responses or tracklets. Flow is modeled as an indicator to link two nodes or not. To meet the flow balance requirement, a source node and a sink node corresponding to the start and the end of a trajectory are added to the graph. One trajectory corresponds to one flow path in the graph. The total flow transited from the source node to the sink node equals to the number of trajectories, and the cost of transition is the negative log-likelihood of all the association hypotheses. Note that the globally optimal solution can be obtained in polynomial time, *e.g.*using the push-relabel algorithm. This model is exceptionally popular and has been widely adopted [40,19,101,45,43,148,147,149,150].

Conditional random field. The conditional random field model is adopted to handle the MOT problem in [62,1,118,151]. Defining a graph $G = (V, E)$ where V is the set of nodes and E is the set of edges, low-level tracklets are given as input to the graph. Each node in the graph represents observations [118] or pairs of tracklets [62], and a label is predicted to indicate which track the observations belongs to or whether to link the tracklets.

MWIS. The maximum-weight independent set (MWIS) is the heaviest subset of non-adjacent nodes of an attributed graph. As in the CRF model described above, nodes in the attribute graph represent pairs of tracklets in successive frames, weights of nodes represent the affinity of the tracklet pair, and the edge is connected if two tracklets share the same detection. Given this graph, the data association is modeled as the MWIS problem [109,48].

3.6.3. Discussion

In practice, deterministic optimization or energy minimization is employed more popularly compared with probabilistic approaches. Although the probabilistic approaches provide a more intuitive and complete solution to the problem, they are usually difficult to infer. On the contrary, energy minimization could obtain a “good enough” solution in a reasonable time.

3.7. Summary

As described above, we have introduced and reviewed different components of an MOT system. It is important to note that not all existing MOT methods have all the components. For example, interaction is not modeled in some studies. Some models are only necessary in specific cases, such as the crowd motion pattern in the case of extremely crowded scenarios. Occlusion is not specifically handled in some of the existing works. In general, appearance, motion and inference

Table 7
An overview of evaluation metrics for MOT. The up arrow (*resp.* down arrow) indicates that the performance is better if the quantity is greater (*resp.* smaller).

Metric	Description	Note	Metric	Description	Note
Recall	Ratio of correctly matched detections to ground-truth detections	↑	TDE	Distance between the ground-truth annotation and the tracking result	↓
Precision	Ratio of correctly matched detections to total result detections	↑	OSPA	Cardinality error, label error and spatial distance between ground truth and the tracking results	↓
FAF/FPPI	Number of false alarms per frame averaged over a sequence	↓	MT	Percentage of ground-truth trajectories which are covered by the tracker output for more than 80% of their length	↑
MODA	Combines missed detections and FAF	↑	ML	Percentage of ground-truth trajectories which are covered by the tracker output for less than 20% of their length	↓
MODP	Average overlap between true positives and ground truth	↑	PT	1.0 - MT - ML	-
MOTA	Combines false negatives, false positives and mismatch rate	↑	FM	Number of times that a ground-truth trajectory is interrupted in the tracking result	↓
IDS	Number of times that a tracked trajectory changes its matched ground-truth identity (or vice versa)	↓	RS	Ratio of tracks which are correctly recovered from short occlusion	↑
MOTP	Overlap between the estimated positions and the ground truth averaged over the matches	↑	RL	Ratio of tracks which are correctly recovered from long occlusion	↑

are mandatory in most methods. Let us take the simplest case as an example, *i.e.*, using a single tracker to track each object individually. In this example, interaction, exclusion and occlusion are not addressed. But appearance and motion models are still necessary with an inference model.

It is also notable that, these components are not orthogonal to each other. They can usually be combined and integrated for satisfactory performance. For instance, the interaction is modeled as several terms along with terms regarding appearance, motion and exclusion modeling, and the resulted objective is optimized with deterministic techniques [85]. Four kinds of features like appearance, motion and location features are concatenated for computing tracklet-object similarity with a two-layer network in [152]. Appearance and motion features are fused by affinity sub-net for more powerful discrimination in [153].

4. MOT evaluation

For a given MOT approach, metrics and datasets are required to evaluate its performance quantitatively. This is important for two reasons. On the one hand, it is essential to measure the influence of different components and parameters on the overall performance to design the best system. On the other hand, it is desirable to have a direct comparison to other methods. Performance evaluation for MOT is not straightforward, as we will see in this section.

Evaluation metrics [154] of MOT approaches are crucial as they provide a means for fair quantitative comparison. A brief review on different MOT evaluation metrics is presented in this section. As many approaches to MOT employ the tracking-by-detection strategy, they often measure detection performance as well as tracking performance. Metrics for object detection are therefore adopted in MOT approaches. Based on this, MOT metrics can be roughly categorized into two sets evaluating detection and tracking respectively, as listed in Table 7.

4.1. Metrics

4.1.1. Metrics for detection

We further group metrics for detection into two subsets. One set measures accuracy, and the other one measures precision.

Accuracy. The commonly used Recall and Precision metrics as well as the average False Alarms per Frame (FAF) rate are employed as MOT metrics [1]. Choi et al. [66] use the False Positive Per Image (FPPI) to evaluate detection performance in MOT. A comprehensive metric called Multiple Object Detection Accuracy (MODA), which considers the relative number of false positives and miss detections is proposed in [155].

Precision. The Multiple Object Detection Precision (MODP) metric measures the quality of alignment between predicted detections and the ground truths [155].

Table 8
Statistics of publicly available datasets. # V and # F mean how many videos and frames are included in the dataset. "Multi-view" and "GT" indicate whether multi-view data and ground truth are provided. "Indoor" and "Outdoor" denote the types of scenarios in the dataset. For the last four items, the check and cross marks mean YES and NO, respectively. The page of a dataset can be accessed by clicking the dataset name.

Dataset	# V	# F	Multi-view	GT	Indoor	Outdoor
MOT16	14	11K	×	✓	✓	✓
KITTI	50	–	✓	✓	×	✓
PETS 2016	13	–	✓	✓	×	✓
PETS 2009	3	–	✓	✓	×	✓
CAVIAR	54	–	✓	✓	✓	×
TUD Stadtmitte	1	179	×	✓	×	✓
TUD Campus	1	71	×	✓	×	✓
TUD Crossing	1	201	×	✓	×	✓
Caltech Pedestrian	137	250K	×	✓	×	✓
UBC Hockey	1	≈100	×	×	×	✓
ETH Pedestrian	8	4K	✓	✓	×	✓
ETHZ Central	3	13K	×	✓	×	✓
Town Centre	1	4.5K	×	✓	×	✓
Zara	4	–	×	×	×	✓
UCSD	98	–	×	×	×	✓
UCF Marathon	3	1.3K	×	✓	×	✓
ParkingLOT	3	2.7K	×	✓	×	✓

4.1.2. Metrics for tracking

Metrics for tracking are classified into four subsets by different attributes as follows.

Accuracy. This kind of metrics measures how accurately an algorithm can track targets. The metric of ID switches (IDs) [85] counts how many times an MOT algorithm switches between objects. The Multiple Object Tracking Accuracy (MOTA) metric [156] combines the false positive rate, false negative rate and mismatch rate into a single number, giving a fairly reasonable quantity for the overall tracking performance. Despite some drawbacks and criticisms, this is by far the most widely accepted evaluation measure for MOT.

Precision. Three metrics, Multiple Object Tracking Precision (MOTP) [156], Tracking Distance Error (TDE) [38] and OSPA [157] belong to this subset. They describe how precisely the objects are tracked measured by bounding box overlap and/or distance. Specifically, cardinality error and label error are additionally considered in [157].

Completeness. Metrics for completeness indicate how completely the ground truth trajectories are tracked. The numbers of Mostly Tracked (MT), Partly Tracked (PT), Mostly Lost (ML) and Fragmentation (FM) [42] belong to this set.

Robustness. To assess the ability of an MOT algorithm to recover from occlusion, metrics called Recover from Short-term occlusion (RS) and Recover from Long-term occlusion (RL) are introduced in [53].

4.2. Datasets

To compare with various existing methods and determine the state of the arts in MOT, publicly available datasets are employed to evaluate the proposed methods in individual publications. Table 8 gives the most popular datasets used in the literature and provides detailed statistics of these datasets.

These datasets have played an important role in the progress of MOT. However, there are some issues with them. First, the scale of datasets for MOT is relatively smaller than that of SOT, *e.g.*, the sequences used in the online object tracking benchmark [30] and the VOT challenge [158], which drive the fast development and standardized evaluation of SOT. Second, current datasets focus on pedestrians. This can be attributed to the fact that pedestrian detection has achieved great success in recent years. However, exciting progress of multi-class detection has been achieved in more recent years. We believe multi-class-multi-object tracking is feasible building upon the detection module of multi-class objects. Thus, it is time to move towards datasets of multi-class objects for MOT.

4.3. Public algorithms

We examine the literature and list algorithms for which the associated source codes are publicly available to make further comparisons convenient in Table 9.

Compared with SOT, there seem to be not many public programs. Admittedly, progress in SOT is larger than that of MOT recently. One reason can be that, many researchers have made their codes publicly available. We here encourage researchers to publish code for research convenience of others in the future.

4.4. Benchmark results

We list public results on the datasets mentioned above to get a direct comparison among different approaches and provide convenience for future comparison. Due to space limitation, we only present results on the most commonly employed

Table 9
A list of publicly available program codes. Please click the reference to access the corresponding code.

Ref.	Note	Ref.	Note
Choi & Savarese [66]	C++	Pirsiavash et al. [43]	MATLAB
Jiang et al. [139]	C	Possegger et al. [159]	MATLAB
Milan et al. [47]	MATLAB	Rodriguez et al. [80]	MATLAB
Andriyenko et al. [50]	MATLAB	Bewley et al. [160]	Python
Milan et al. [118]	MATLAB	Kim et al. [161]	MATLAB
Zamir et al. [162]	MATLAB	Dicle et al. [70]	MATLAB
Berclaz et al. [44]	C	Bae & Yoon [163]	MATLAB
Rezatofighi et al. [164]	MATLAB	Xiang et al. [59]	MATLAB
Zhang & Laurens [55,56]	MATLAB/C	Solera et al. [165]	MATLAB

Table 10
Quantitative results on the PETS2009-S2L1 dataset, extracted from the respective publications. Methods, ordered by year of publication, are labeled with Online and Offline tags. Concerning each metric, the best results are shown in bold. The table shows that progress has been achieved while there seems not much research attention in this topic recently.

Ref.	MOTA ↑	MOTP ↑	IDS ↓	Pre. ↑	Rec. ↑	MT ↑	PT	ML ↓	FM ↓	F1 ↑	Year	Note	Source
Berclaz et al. [140]	0.830	0.520	-	0.820	0.530	-	-	-	-	0.644	2009	Off	[81]
Yang et al. [166]	0.759	0.538	-	-	-	-	-	-	-	-	2009	On	[159]
Alahi et al. [167]	0.830	0.520	-	0.690	0.530	-	-	-	-	0.600	2009	On	[81]
Conte et al. [168]	0.810	0.570	-	0.850	0.580	-	-	-	-	0.690	2010	On	[81]
Berclaz et al. [44]	0.803	0.720	13	0.963	0.838	0.739	0.174	0.087	22	0.896	2011	Off	[169]
Shitrit et al. [170]	0.815	0.584	19	0.907	0.908	-	-	-	-	0.907	2011	Off	[162]
Andriyenko & Schindler [122]	0.863	0.787	38	0.976	0.895	0.783	0.174	0.043	21	0.934	2011	Off	[169]
Henriques et al. [64]	0.848	0.687	10	0.924	0.940	-	-	-	-	0.932	2011	Off	[162]
Pirsiavash et al. [43]	0.774	0.743	57	0.972	0.812	0.609	0.347	0.043	62	0.885	2011	Off	[169]
Kuo et al. [111]	-	-	1	**0.996**	0.895	0.789	0.211	**0.000**	23	0.943	2011	Off	[49]
Andriyenko et al. [129]	0.917	0.745	11	-	-	-	-	-	-	-	2011	Off	[58]
Leal et al. [171]	0.670	-	-	-	-	-	-	-	-	-	2011	Off	[58]
Breitenstein et al. [172]	0.797	0.563	-	-	-	-	-	-	-	-	2011	On	[159]
Izadinia et al. [81]	0.907	0.760	-	0.968	0.952	-	-	-	-	0.960	2012	Off	[81]
Zamir et al. [162]	0.903	0.690	8	0.936	0.965	-	-	-	-	0.950	2012	Off	[162]
Andriyenko et al. [50]	0.883	0.796	18	0.987	0.900	0.826	0.174	**0.000**	14	0.941	2012	Off	[169]
Yang & Nevatia [49]	-	-	**0**	0.990	0.918	0.895	0.105	**0.000**	9	0.953	2012	Off	[49]
Yang & Nevatia [51]	-	-	**0**	0.948	0.978	0.950	0.050	**0.000**	**2**	0.963	2012	Off	[173]
Leal et al. [174]	0.760	0.600	-	-	-	-	-	-	-	-	2012	Off	[159]
Zhang et al. [58]	0.933	0.682	19	-	-	-	-	-	-	-	2012	On	[58]
Segal & Reid [175]	0.900	0.750	6	-	-	0.890	-	-	-	-	2013	Off	[176]
Kumar & Vleeschouwer [119]	0.910	0.700	5	-	-	-	-	-	-	-	2013	Off	[176]
Hofman et al. [177]	0.980	0.828	10	-	-	**1.000**	0.000	**0.000**	11	-	2013	Off	[159]
Milan et al. [118]	0.903	0.743	22	-	-	0.783	0.217	**0.000**	15	-	2013	Off	[118]
Hofmann et al. [178]	0.978	0.753	8	0.991	**0.990**	**1.000**	0.000	**0.000**	8	**0.990**	2013	Off	[178]
Shi et al. [46]	0.927	0.818	7	0.982	0.960	0.947	0.053	**0.000**	11	0.971	2013	Off	[179]
Wu et al. [180]	0.928	0.743	8	-	-	**1.000**	0.000	**0.000**	11	-	2013	On	[180]
Milan et al. [47]	0.906	0.802	11	0.984	0.924	0.913	0.043	0.043	-	0.953	2014	Off	[169]
Wen et al. [169]	0.927	0.729	5	0.984	0.944	0.957	0.043	**0.000**	10	0.964	2014	Off	[169]
Shi et al. [179]	0.961	0.818	4	0.989	0.977	0.947	0.053	**0.000**	6	0.983	2014	Off	[179]
Bae & Yoon [163]	0.830	0.696	4	-	-	**1.000**	0.000	**0.000**	4	-	2014	On	[163]
Possegger et al. [159]	**0.981**	0.805	9	-	-	**1.000**	0.000	**0.000**	16	-	2014	On	[159]
Zhang et al. [173]	0.956	**0.916**	**0**	0.986	0.970	0.950	0.050	**0.000**	4	0.978	2015	Off	[173]
Dehghan et al. [176]	0.904	0.631	3	-	-	0.950	0.050	**0.000**	-	-	2015	Off	[176]
Lenz et al. [148]	0.890	0.870	7	-	-	0.890	0.110	0	100	-	2015	On	[148]

PETS2009-S2L1 sequence in Table 10. Results of other datasets are present in the supplementary material. Please note that this kind of direct comparison on the same dataset may not be fair due to the following points:

- Different methodologies. For example, some publications belong to offline methods while others belong to online ones. Due to the difference described in Section 2.2.2, it is unfair to directly compare them because the former have access to much more information.
- Different detection hypotheses. Different approaches adopt various detectors to obtain detection hypotheses. One approach based on different detection hypotheses would output different results, let alone different approaches.
- Some approaches aggregate observations from multiple views while others utilize information from a single view. This makes a direct comparison between them difficult.

Table 11
Benchmark result comparison between offline and online methods on the PETS2009-S2L1 dataset.

Proc.	MOTA↑	MOTP↑	MT↑	ML↓	IDS↓	FM↓
offline	0.935	0.775	0.95	0	5.9	8.3
online	0.913	0.748	1.00	0	7.0	10.3

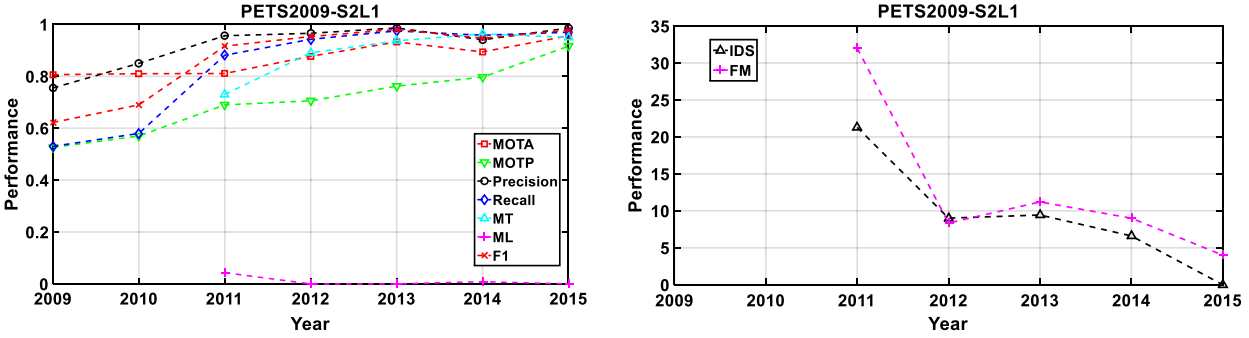

Fig. 6. Statistics of results in different years on the PETS2009-S2L1 dataset in terms of MOTA, MOTP, Precision, Recall, MT, ML, F1 metrics (left), IDS and FM metrics (right).

- Prior information, such as scene structure and the number of pedestrians, is exploited by some approaches, making a direct quantitative comparison to other methods which do not use that information questionable.

Strictly speaking, in order to make a direct and fair comparison, one needs to fix all the other components while varying the one under consideration. For instance, adopting different data association models while keeping all other parts the same could directly compare performance of different data association methods. This is the main goal of recent MOT benchmarks like KITTI [181] and MOTChallenge [31,182], which specifically focus on a centralized evaluation of multiple object tracking. For intensive experimental comparison among different MOT solutions, please refer the respective benchmarks. In spite of the issues mentioned above, it is still worthy to list all the public results on the same dataset or sequence due to the following reasons.

- By compiling all published results into a single table, it at least provides an intuitive comparison among different methods on the same dataset and a convenience for future works.
- Although this particular comparison among individual methods may not be fair, an approximate comparison between different types of methods such as that between offline and online methods could tell us how these types of methods perform on public datasets.
- Additionally, we could observe how the research of MOT progresses over time by comparing performance of methods across years.

We report the results in terms of the MOTA, MOTP, IDS, Precision, Recall, MT, PT, ML, FM, and F1 metrics. At the same time, we tag the results with online and offline labels. Please note that: 1) there are missing entries, because we did not find the corresponding value neither from the original publication nor from other publications which cite it, and 2) in some cases, there could be different results for a unique publication (for example, results from the original publication versus results from another publication which compares with it). This discrepancy may arise because different configurations are adopted (*e.g.* different detection hypotheses). In this case, we quote the most popularly cited one.

We conduct an analysis of benchmark results on the PETS2009-S2L1 dataset to investigate the comparison between offline methods and online methods. Choosing results from publications that appeared in 2012 and later, we average values of each metric across each type of methods, and report the mean value in Table 11. As expected, offline methods generally outperform online ones *w.r.t.* most metrics. This is due to the fact that offline methods employ global temporal information for the estimation.

Additionally, we analyze the evaluation results on the PETS2009-S2L1 dataset over time. Specifically, we plot the metric values of methods in each year ranging from 2009 to 2015 in Fig. 6. It is no surprise that the performance improves over the years. We suspect that factors such as better models and progress in object detection [183–190] all contribute to the achieved progress. It should also be noted that a research community focuses on a specific dataset over time and certain methods may be a result of "over-fitting" to that dataset as opposed to general progress towards solving the problem.

5. Summary

This paper has described methods and problems related to the task of Multiple Object Tracking (MOT) in videos. As the first comprehensive literature review in the past decade, it has presented a unified problem formulation and several ways of categorization of existing methods, described the key components within state-of-the-art MOT approaches, and

discussed the evaluations of MOT algorithms including evaluation metrics, public datasets, open source implementations, and benchmark results. Although great progress in MOT has been made in the past decades, there still remain several issues in the current MOT research and many open problems to be studied.

5.1. Existing issues

We have discussed the existing issues of datasets (Section 4.2) and public algorithms (Section 4.3). Except these issues, there are still some remarkable ones as follows:

One major issue in the MOT research is that, performance of an MOT method depends heavily on the object detectors. For example, the widely used tracking-by-detection paradigm is built upon an object detector, which provides detection hypotheses to drive the tracking procedure. Given different sets of detection hypotheses while fixing other components, an identical approach would produce tracking results with significant performance differences. In the community, sometimes no description about the detection module is given in the approach. This makes the comparisons with other approaches infeasible. Established benchmarks like KITTI and MOTChallenge attempt to alleviate this problem and are also moving towards a more principled and unified evaluation of detection and tracking (cf. MOT17).

Another nuisance is that, when developing an MOT solution, there are many parameters if this algorithm is too complicated. This leads to a difficulty of tuning the method. Meanwhile, it is also difficult for others to implement the approach and reproduce the result.

Some approaches perform well in specific video sequences. While applied to other cases, however, they may not produce satisfying results. The reasons are multi-fold. Differences of camera view, or the status of camera (moving versus static) can lead to this issue. It might also be caused by the fact that object detectors utilized by MOT approaches are trained in specific videos and do not generalize well in other video sequences.

All these issues restrict further development of the MOT research and its applications in practical systems. Recently, attempts to deal with some of these issues have been made, *e.g.*, the MOT Benchmark [182] provides a large set of annotated testing video sequences, unified detection hypotheses, standard evaluation tools, *etc.*This is very likely to advance the further studies and developments of MOT techniques.

5.2. Future directions

Even after decades of research on the MOT problem, there are still numerous research opportunities in studying this problem. Here we would like to point out some more prevalent problems and provide possible research directions.

MOT with video adaptation. As mentioned above, the majority of current MOT methods requires an offline trained object detector. There arises a problem that the detection result for a specific video is not optimal since the object detector is not trained for the given video. This often limits the performance of multiple object tracking. A customization of the object detector is necessary to improve MOT performance. One solution proposed by Shu et al. [191] adapts a generic pedestrian detector to a specific video by progressively refining the generic pedestrian detector. This is one important direction to follow in order to improve the pre-processing stage for MOT methods.

MOT under multiple cameras. It is obvious that MOT would benefit from multi-camera settings [192,193]. There are two kinds of configurations of multiple cameras. The first one is that multiple cameras record the same scene, *i.e.*, multiple views. However, one key question in this setting is how to fuse the information from multiple cameras. The second one is that each camera records a different scene, *i.e.*, a non-overlapping multi-camera network. In that case, the data association across multiple cameras becomes a re-identification problem.

Multiple 3D object tracking. Most of the current approaches focus on multiple object tracking in 2D, *i.e.*, on the image plane, even in the case of multiple cameras. 3D tracking [194], which could provide more accurate position, size estimation and effective occlusion handling for high-level computer vision tasks, is potentially more useful. However, 3D tracking requires camera calibration, or has to overcome other challenges for estimating camera poses and scene layout. Meanwhile, 3D model design is another issue exclusive to 2D MOT.

MOT with scene understanding. Previous studies [37,195,196] have been performed to analyze over-crowded scenarios such as underground train stations during peak hours and demonstrations in public places. In this kind of scenarios, most objects are small and/or largely occluded, thus are very difficult to track. The analyzing results from scene understanding can provide contextual information and scene structure, which is very helpful to the tracking problem if it is better incorporated into an MOT algorithm.

MOT with deep learning. Deep learning based models have emerged as an extremely powerful framework to deal with different kinds of vision problems including image classification [197], object detection [185–187], and more relevantly single object tracking [183]. For the MOT problem, the strong observation model provided by the deep learning model for target detection can boost the tracking performance significantly [198,199]. The formulation and modeling of the target association problem using deep neural networks [200–203] need more research efforts, although the first attempt to employ sequential neural networks for online MOT has been made very recently. The modules like attention mechanism [204], LSTM [114,97] are also employed by researchers for solving the MOT problem.

MOT with other computer vision tasks. Although multiple object tracking serves other high-level computer vision tasks, there is a trend to solve multi-object tracking with some other computer vision tasks jointly as they are beneficial to each

other. Possible combinations include object segmentation [205–208], re-identification [209,193,210], human pose estimation [18,211–214], and action recognition [19].

Besides the above future directions, since the current MOT research is mainly focused on tracking multiple humans in a surveillance scenario, the extensions of the current MOT research to other types of targets (*e.g.*, vehicles, animals, *etc.*) and scenarios (*e.g.*, traffic scenes, aerial photographs, *etc.*) are also very good research directions, since the problem settings and difficulties for tracking different types of targets under different scenarios are sometimes quite different from those in tracking multiple humans in a surveillance scenario.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported in part by the National Key Research and Development Program of China under Grant No. 2019AAA010340X, in part by the Natural Science Foundation of China under Grant No. 62076238, and in part by the Strategic Priority Research Program of Chinese Academy of Sciences under Grant No. XDA27000000.

References

[1] B. Yang, C. Huang, R. Nevatia, Learning affinities and dependencies for multi-target tracking using a CRF model, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Colorado Springs, CO, USA, 2011, pp. 1233–1240.
[2] S. Pellegrini, A. Ess, K. Schindler, L. Van Gool, You'll never walk alone: modeling social behavior for multi-target tracking, in: Proc. IEEE Int. Conf. Comput. Vis., Kyoto, Japan, 2009, pp. 261–268.
[3] D. Koller, J. Weber, J. Malik, Robust multiple car tracking with occlusion reasoning, in: Proc. Eur. Conf. Comput. Vis., Stockholm, Sweden, 1994, pp. 189–196.
[4] M. Betke, E. Haritaoglu, L.S. Davis, Real-time multiple vehicle detection and tracking from a moving vehicle, Mach. Vis. Appl. 12 (2) (2000) 69–83.
[5] W.-L. Lu, J.-A. Ting, J. Little, K. Murphy, Learning to track and identify players from broadcast sports videos, IEEE Trans. Pattern Anal. Mach. Intell. 35 (7) (2013) 1704–1716.
[6] J. Xing, H. Ai, L. Liu, S. Lao, Multiple player tracking in sports video: a dual-mode two-way bayesian inference approach with progressive observation modeling, IEEE Trans. Image Process. 20 (6) (2011) 1652–1667.
[7] P. Nillius, J. Sullivan, S. Carlsson, Multi-target tracking-linking identities using bayesian network inference, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2, New York City, NY, USA, 2006, pp. 2187–2194.
[8] W. Luo, T.-K. Kim, B. Stenger, X. Zhao, R. Cipolla, Bi-label propagation for generic multiple object tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Columbus, OH, USA, 2014, pp. 1290–1297.
[9] M. Betke, D.E. Hirsh, A. Bagchi, N.I. Hristov, N.C. Makris, T.H. Kunz, Tracking large variable numbers of objects in clutter, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Minneapolis, MN, USA, 2007, pp. 1–8.
[10] Z. Khan, T. Balch, F. Dellaert, An MCMC-based particle filter for tracking multiple interacting targets, in: Proc. Eur. Conf. Comput. Vis., Prague, Czech Republic, 2004, pp. 279–290.
[11] C. Spampinato, Y.-H. Chen-Burger, G. Nadarajan, R.B. Fisher, Detecting, tracking and counting fish in low quality unconstrained underwater videos, in: Proc. Int. Conf. Comput. Vis. Theory Appl., Madeira, Portugal, 2008, pp. 514–519.
[12] C. Spampinato, S. Palazzo, D. Giordano, I. Kavasidis, F.-P. Lin, Y.-T. Lin, Covariance based fish tracking in real-life underwater environment, in: Proc. Int. Conf. Comput. Vis. Theory Appl., Rome, Italy, 2012, pp. 409–414.
[13] E. Fontaine, A.H. Barr, J.W. Burdick, Model-based tracking of multiple worms and fish, in: Proc. IEEE Int. Conf. Comput. Vis. Workshops, Rio de Janeiro, Brazil, 2007, pp. 1–8.
[14] E. Meijering, O. Dzyubachyk, I. Smal, W.A. van Cappellen, Tracking in cell and developmental biology, Semin. Cell Dev. Biol. 20 (8) (2009) 894–902.
[15] K. Li, E.D. Miller, M. Chen, T. Kanade, L.E. Weiss, P.G. Campbell, Cell population tracking and lineage construction with spatiotemporal context, Med. Image Anal. 12 (5) (2008) 546–566.
[16] K. Bozek, L. Hebert, A.S. Mikheyev, G.J. Stephens, Towards dense object tracking in a 2D honeybee hive, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Salt Lake City, UT, USA, 2018, pp. 4185–4193.
[17] G. Duan, H. Ai, S. Cao, S. Lao, Group tracking: exploring mutual relations for multiple object tracking, in: Proc. Eur. Conf. Comput. Vis., Florence, Italy, 2012, pp. 129–143.
[18] T. Pfister, J. Charles, A. Zisserman, Flowing ConvNets for human pose estimation in videos, in: Proc. IEEE Int. Conf. Comput. Vis., Santiago, Chile, 2015, pp. 1913–1921.
[19] W. Choi, S. Savarese, A unified framework for multi-target tracking and collective activity recognition, in: Proc. Eur. Conf. Comput. Vis., Florence, Italy, 2012, pp. 215–230.
[20] W. Hu, T. Tan, L. Wang, S. Maybank, A survey on visual surveillance of object motion and behaviors, IEEE Trans. Syst. Man Cybern., Part C, Appl. Rev. 34 (3) (2004) 334–352.
[21] X. Wang, Intelligent multi-camera video surveillance: a review, Pattern Recognit. Lett. 34 (1) (2013) 3–19.
[22] J. Candamo, M. Shreve, D.B. Goldgof, D.B. Sapper, R. Kasturi, Understanding transit scenes: a survey on human behavior-recognition algorithms, IEEE Trans. Intell. Transp. Syst. 11 (1) (2010) 206–224.
[23] H. Uchiyama, E. Marchand, Object detection and pose tracking for augmented reality: recent approaches, in: Proc. Korea-Japan Joint Workshop on Frontiers of Computer Vision, Kawasaki, Japan, 2012, pp. 1–8.
[24] B. Zhan, D.N. Monekosso, P. Remagnino, S.A. Velastin, L.-Q. Xu, Crowd analysis: a survey, Mach. Vis. Appl. 19 (5–6) (2008) 345–357.
[25] I.S. Kim, H.S. Choi, K.M. Yi, J.Y. Choi, S.G. Kong, Intelligent visual surveillance-a survey, Int. J. Control. Autom. Syst. 8 (5) (2010) 926–939.
[26] D.A. Forsyth, O. Arikan, L. Ikemoto, J. O'Brien, D. Ramanan, et al., Computational studies of human motion: part 1, tracking and motion synthesis, Found. Trends Comput. Graph. Vis. 1 (2–3) (2006) 77–254.
[27] K. Cannons, A review of visual tracking, Tech. Rep. CSE-2008-07, Dept. Comput. Sci. Eng., York Univ., 1991.
[28] A. Yilmaz, O. Javed, M. Shah, Object tracking: a survey, ACM Comput. Surv. 38 (4) (2006) 13.

[29] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, A.V.D. Hengel, A survey of appearance models in visual object tracking, ACM Trans. Intell. Syst. Technol. 4 (4) (2013) 58.
[30] Y. Wu, J. Lim, M.-H. Yang, Online object tracking: a benchmark, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Anchorage, AL, USA, 2013, pp. 2411–2418.
[31] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, K. Schindler, MOTChallenge 2015: towards a benchmark for multi-target tracking, arXiv:1504.01942, http://arxiv.org/abs/1504.01942.
[32] Z.-Q. Zhao, P. Zheng, S.-t. Xu, X. Wu, Object detection with deep learning: a review, IEEE Trans. Neural Netw. Learn. Syst. 30 (11) (2019) 3212–3232.
[33] J. Xing, H. Ai, S. Lao, Multi-object tracking through occlusions by local tracklets filtering and global tracklets association with detection responses, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Miami, FL, USA, 2009, pp. 1200–1207.
[34] M.D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, L. Van Gool, Robust tracking-by-detection using a detector confidence particle filter, in: Proc. IEEE Int. Conf. Comput. Vis., Kyoto, Japan, 2009, pp. 1515–1522.
[35] M. Yang, F. Lv, W. Xu, Y. Gong, Detection driven adaptive multi-cue integration for multiple human tracking, in: Proc. IEEE Int. Conf. Comput. Vis., Kyoto, Japan, 2009, pp. 1554–1561.
[36] D. Mitzel, B. Leibe, Real-time multi-person tracking with detector assisted structure propagation, in: Proc. IEEE Int. Conf. Comput. Vis. Workshops, Barcelona, Spain, 2011, pp. 974–981.
[37] M. Rodriguez, J. Sivic, I. Laptev, J.-Y. Audibert, Data-driven crowd analysis in videos, in: Proc. IEEE Int. Conf. Comput. Vis., Barcelona, Spain, 2011, pp. 1235–1242.
[38] L. Kratz, K. Nishino, Tracking with local spatio-temporal motion patterns in extremely crowded scenes, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., San Francisco, CA, USA, 2010, pp. 693–700.
[39] D. Reid, An algorithm for tracking multiple targets, IEEE Trans. Autom. Control 24 (6) (1979) 843–854.
[40] L. Zhang, Y. Li, R. Nevatia, Global data association for multi-object tracking using network flows, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Anchorage, AL, USA, 2008, pp. 1–8.
[41] C. Huang, B. Wu, R. Nevatia, Robust object tracking by hierarchical association of detection responses, in: Proc. Eur. Conf. Comput. Vis., Marseille, France, 2008, pp. 788–801.
[42] Y. Li, C. Huang, R. Nevatia, Learning to associate: HybridBoosted multi-target tracker for crowded scene, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Miami, FL, USA, 2009, pp. 2953–2960.
[43] H. Pirsiavash, D. Ramanan, C.C. Fowlkes, Globally-optimal greedy algorithms for tracking a variable number of objects, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Colorado Springs, CO, USA, 2011, pp. 1201–1208.
[44] J. Berclaz, F. Fleuret, E. Turetken, P. Fua, Multiple object tracking using k-shortest paths optimization, IEEE Trans. Pattern Anal. Mach. Intell. 33 (9) (2011) 1806–1819.
[45] A. Butt, R. Collins, Multi-target tracking by lagrangian relaxation to min-cost network flow, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Portland, OR, USA, 2013, pp. 1846–1853.
[46] X. Shi, H. Ling, J. Xing, W. Hu, Multi-target tracking by rank-1 tensor approximation, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Portland, OR, USA, 2013, pp. 2387–2394.
[47] A. Milan, S. Roth, K. Schindler, Continuous energy minimization for multitarget tracking, IEEE Trans. Pattern Anal. Mach. Intell. 36 (1) (2014) 58–72.
[48] W. Brendel, M. Amer, S. Todorovic, Multiobject tracking as maximum weight independent set, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Colorado Springs, CO, USA, 2011, pp. 1273–1280.
[49] B. Yang, R. Nevatia, Multi-target tracking by online learning of non-linear motion patterns and robust appearance models, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Providence, RI, USA, 2012, pp. 1918–1925.
[50] A. Andriyenko, K. Schindler, S. Roth, Discrete-continuous optimization for multi-target tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Providence, RI, USA, 2012, pp. 1926–1933.
[51] B. Yang, R. Nevatia, Online learned discriminative part-based appearance models for multi-human tracking, in: Proc. Eur. Conf. Comput. Vis., Florence, Italy, 2012, pp. 484–498.
[52] B. Bose, X. Wang, E. Grimson, Multi-class object tracking algorithm that handles fragmentation and grouping, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Minneapolis, MN, USA, 2007, pp. 1–8.
[53] B. Song, T.-Y. Jeng, E. Staudt, A.K. Roy-Chowdhury, A stochastic graph evolution framework for robust multi-target tracking, in: Proc. Eur. Conf. Comput. Vis., Hersonissos, Greece, 2010, pp. 605–619.
[54] W. Hu, X. Li, W. Luo, X. Zhang, S. Maybank, Z. Zhang, Single and multiple object tracking using log-Euclidean Riemannian subspace and block-division appearance model, IEEE Trans. Pattern Anal. Mach. Intell. 34 (12) (2012) 2420–2440.
[55] L. Zhang, L. van der Maaten, Structure preserving object tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Portland, OR, USA, 2013, pp. 1838–1845.
[56] L. Zhang, L. van der Maaten, Preserving structure in model-free tracking, IEEE Trans. Pattern Anal. Mach. Intell. 36 (4) (2014) 756–769.
[57] M. Yang, T. Yu, Y. Wu, Game-theoretic multiple target tracking, in: Proc. IEEE Int. Conf. Comput. Vis., Rio de Janeiro, Brazil, 2007, pp. 1–8.
[58] J. Zhang, L.L. Presti, S. Sclaroff, Online multi-person tracking by tracker hierarchy, in: Proc. IEEE Int. Conf. Adv. Video Signal-based Surveillance, Beijing, China, 2012, pp. 379–385.
[59] Y. Xiang, A. Alahi, S. Savarese, Learning to track: online multi-object tracking by decision making, in: Proc. IEEE Int. Conf. Comput. Vis., Santiago, Chile, 2015, pp. 4705–4713.
[60] J. Hong Yoon, C.-R. Lee, M.-H. Yang, K.-J. Yoon, Online multi-object tracking via structural constraint event aggregation, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Las Vegas, NV, USA, 2016, pp. 1392–1400.
[61] Z. Qin, C.R. Shelton, Improving multi-target tracking via social grouping, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Providence, RI, USA, 2012, pp. 1972–1978.
[62] B. Yang, R. Nevatia, An online learned CRF model for multi-target tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Providence, RI, USA, 2012, pp. 2034–2041.
[63] C.-H. Kuo, C. Huang, R. Nevatia, Multi-target tracking by on-line learned discriminative appearance models, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., San Francisco, CA, USA, 2010, pp. 685–692.
[64] J.F. Henriques, R. Caseiro, J. Batista, Globally optimal solution to multi-object tracking with merged measurements, in: Proc. IEEE Int. Conf. Comput. Vis., Barcelona, Spain, 2011, pp. 2470–2477.
[65] D. Sugimura, K.M. Kitani, T. Okabe, Y. Sato, A. Sugimoto, Using individuality to track individuals: clustering individual trajectories in crowds using local appearance and frequency trait, in: Proc. IEEE Int. Conf. Comput. Vis., Kyoto, Japan, 2009, pp. 1467–1474.
[66] W. Choi, S. Savarese, Multiple target tracking in world coordinate with single, minimally calibrated camera, in: Proc. Eur. Conf. Comput. Vis., Hersonissos, Greece, 2010, pp. 553–567.
[67] S. Hong, S. Kwak, B. Han, Orderless tracking through model-averaged posterior estimation, in: Proc. IEEE Int. Conf. Comput. Vis., Sydney, Australia, 2013, pp. 2296–2303.
[68] W. Luo, T.-K. Kim, Generic object crowd tracking by multi-task learning, in: Proc. British Mach. Vis. Conf., Bristol, UK, 2013, pp. 1–13.

[69] V. Reilly, H. Idrees, M. Shah, Detection and tracking of large number of targets in wide area surveillance, in: Proc. Eur. Conf. Comput. Vis., Hersonissos, Greece, 2010, pp. 186–199.
[70] C. Dicle, M. Sznaier, O. Camps, The way they move: tracking multiple targets with similar appearance, in: Proc. IEEE Int. Conf. Comput. Vis., Sydney, Australia, 2013, pp. 2304–2311.
[71] G. Brostow, R. Cipolla, Unsupervised bayesian detection of independent motion in crowds, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 1, New York City, NY, USA, 2006, pp. 594–601.
[72] T. Sekii, Robust, real-time 3D tracking of multiple objects with similar appearances, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Las Vegas, NV, USA, 2016, pp. 4275–4283.
[73] S. Ali, M. Shah, Floor fields for tracking in high density crowd scenes, in: Proc. Eur. Conf. Comput. Vis., Marseille, France, 2008, pp. 1–14.
[74] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., San Diego, CA, USA, 2005, pp. 1–6.
[75] D. Mitzel, E. Horbert, A. Ess, B. Leibe, Multi-person tracking with sparse detection and continuous segmentation, in: Proc. Eur. Conf. Comput. Vis., Hersonissos, Greece, 2010, pp. 397–410.
[76] K. Okuma, A. Taleghani, N. De Freitas, J.J. Little, D.G. Lowe, A boosted particle filter: multitarget detection and tracking, in: Proc. Eur. Conf. Comput. Vis., Prague, Czech Republic, 2004, pp. 28–39.
[77] J. Shi, C. Tomasi, Good features to track, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Seattle, WA, USA, 1994, pp. 593–600.
[78] X. Zhao, D. Gong, G. Medioni, Tracking using motion patterns for very crowded scenes, in: Proc. Eur. Conf. Comput. Vis., Florence, Italy, 2012, pp. 315–328.
[79] B. Benfold, I. Reid, Stable multi-target tracking in real-time surveillance video, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Colorado Springs, CO, USA, 2011, pp. 3457–3464.
[80] M. Rodriguez, S. Ali, T. Kanade, Tracking in unstructured crowded scenes, in: Proc. IEEE Int. Conf. Comput. Vis., Kyoto, Japan, 2009, pp. 1389–1396.
[81] H. Izadinia, I. Saleemi, W. Li, M. Shah, (MP)2T: multiple people multiple parts tracker, in: Proc. Eur. Conf. Comput. Vis., Florence, Italy, 2012, pp. 100–114.
[82] S. Walk, N. Majer, K. Schindler, B. Schiele, New features and insights for pedestrian detection, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., San Francisco, CA, USA, 2010, pp. 1030–1037.
[83] W. Choi, Near-online multi-target tracking with aggregated local flow descriptor, in: Proc. IEEE Int. Conf. Comput. Vis., Santiago, Chile, 2015, pp. 3029–3037.
[84] H. Yu, Y. Zhou, J. Simmons, C.P. Przybyla, Y. Lin, X. Fan, Y. Mi, S. Wang, Groupwise tracking of crowded similar-appearance targets from low-continuity image sequences, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Las Vegas, NV, USA, 2016, pp. 952–960.
[85] K. Yamaguchi, A.C. Berg, L.E. Ortiz, T.L. Berg, Who are you with and where are you going?, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Colorado Springs, CO, USA, 2011, pp. 1345–1352.
[86] T. Yu, Y. Wu, N.O. Krahnstoever, P.H. Tu, Distributed data association and filtering for multiple target tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Anchorage, AL, USA, 2008, pp. 1–8.
[87] F. Porikli, O. Tuzel, P. Meer, Covariance tracking using model update based on lie algebra, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 1, New York City, NY, USA, 2006, pp. 728–735.
[88] O. Tuzel, F. Porikli, P. Meer, Region covariance: a fast descriptor for detection and classification, in: Proc. Eur. Conf. Comput. Vis., Graz, Austrian, 2006, pp. 589–600.
[89] A. Ess, B. Leibe, K. Schindler, L. Van Gool, Robust multiperson tracking from a mobile platform, IEEE Trans. Pattern Anal. Mach. Intell. 31 (10) (2009) 1831–1846.
[90] A. Ess, B. Leibe, L. Van Gool, Depth and appearance for mobile scene analysis, in: Proc. IEEE Int. Conf. Comput. Vis., Rio de Janeiro, Brazil, 2007, pp. 1–8.
[91] A. Ess, B. Leibe, K. Schindler, L. Van Gool, A mobile vision system for robust multi-person tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Anchorage, AL, USA, 2008, pp. 1–8.
[92] D.M. Gavrila, S. Munder, Multi-cue pedestrian detection and tracking from a moving vehicle, Int. J. Comput. Vis. 73 (1) (2007) 41–59.
[93] F. Fleuret, J. Berclaz, R. Lengagne, P. Fua, Multicamera people tracking with a probabilistic occupancy map, IEEE Trans. Pattern Anal. Mach. Intell. 30 (2) (2008) 267–282.
[94] Y. Xu, B. Ma, R. Huang, L. Lin, Person search in a scene by jointly modeling people commonness and person uniqueness, in: Proc. ACM Int. Conf. Multimedia, Orlando, Florida, USA, 2014, pp. 937–940.
[95] Y. Xu, X. Liu, Y. Liu, S.-C. Zhu, Multi-view people tracking via hierarchical trajectory composition, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Las Vegas, NV, USA, 2016, pp. 4256–4265.
[96] Y. Xu, L. Qin, X. Liu, J. Xie, S.-C. Zhu, A causal and-or graph model for visibility fluent reasoning in tracking interacting objects, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Salt Lake City, UT, USA, 2018, pp. 2178–2187.
[97] C. Kim, F. Li, J.M. Rehg, Multi-object tracking with neural gating using bilinear LSTM, in: Proc. Eur. Conf. Comput. Vis., Munich, German, 2018, pp. 200–215.
[98] Z. He, J. Li, D. Liu, H. He, D. Barber, Tracking by animation: unsupervised learning of multi-object attentive trackers, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Long Beach, CA, USA, 2019, pp. 1318–1327.
[99] W. Zhang, H. Zhou, S. Sun, Z. Wang, J. Shi, C.C. Loy, Robust multi-modality multi-object tracking, in: Proc. IEEE Int. Conf. Comput. Vis., Seoul, Korea, 2019, pp. 2365–2374.
[100] P.F. Felzenszwalb, D.P. Huttenlocher, Efficient belief propagation for early vision, Int. J. Comput. Vis. 70 (1) (2006) 41–54.
[101] Z. Wu, A. Thangali, S. Sclaroff, M. Betke, Coupling detection and data association for multiple object tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Providence, RI, USA, 2012, pp. 1948–1955.
[102] B. Leibe, K. Schindler, N. Cornelis, L. Van Gool, Coupled object detection and tracking from static cameras and moving vehicles, IEEE Trans. Pattern Anal. Mach. Intell. 30 (10) (2008) 1683–1698.
[103] L. Ren, J. Lu, Z. Wang, Q. Tian, J. Zhou, Collaborative deep reinforcement learning for multi-object tracking, in: Proc. Eur. Conf. Comput. Vis., Munich, German, 2018, pp. 586–602.
[104] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2, New York City, NY, USA, 2006, pp. 2169–2178.
[105] Y. Liu, H. Li, Y.Q. Chen, Automatic tracking of a large number of moving targets in 3D, in: Proc. Eur. Conf. Comput. Vis., Florence, Italy, 2012, pp. 730–742.
[106] V. Takala, M. Pietikainen, Multi-object tracking using color, texture and motion, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Minneapolis, MN, USA, 2007, pp. 1–7.
[107] J. Giebel, D.M. Gavrila, C. Schnörr, A bayesian framework for multi-cue 3D object tracking, in: Proc. Eur. Conf. Comput. Vis., Prague, Czech Republic, 2004, pp. 241–252.
[108] J. Berclaz, F. Fleuret, P. Fua, Robust people tracking with global trajectory optimization, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 1, New York City, NY, USA, 2006, pp. 744–750.

[109] K. Shafique, M.W. Lee, N. Haering, A rank constrained continuous formulation of multi-frame multi-target tracking problem, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Anchorage, AL, USA, 2008, pp. 1–8.
[110] Q. Yu, G. Medioni, I. Cohen, Multiple target tracking using spatio-temporal Markov chain Monte Carlo data association, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Minneapolis, MN, USA, 2007, pp. 1–8.
[111] C.-H. Kuo, R. Nevatia, How does person identity recognition help multi-person tracking?, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Colorado Springs, CO, USA, 2011, pp. 1217–1224.
[112] D. Helbing, P. Molnar, Social force model for pedestrian dynamics, Phys. Rev. E 51 (5) (1995) 4282.
[113] M. Hu, S. Ali, M. Shah, Detecting global motion patterns in complex videos, in: Proc. Int. Conf. Pattern Recognit., Tampa, USA, 2008, pp. 1–5.
[114] A. Maksai, P. Fua, Eliminating exposure bias and metric mismatch in multiple object tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Long Beach, CA, USA, 2019, pp. 4639–4648.
[115] P. Scovanner, M.F. Tappen, Learning pedestrian dynamics from the real world, in: Proc. IEEE Int. Conf. Comput. Vis., Kyoto, Japan, 2009, pp. 381–388.
[116] S. Pellegrini, A. Ess, L. Van Gool, Improving data association by joint modeling of pedestrian trajectories and groupings, in: Proc. Eur. Conf. Comput. Vis., Hersonissos, Greece, 2010, pp. 452–465.
[117] L. Kratz, K. Nishino, Tracking pedestrians using local spatio-temporal motion patterns in extremely crowded scenes, IEEE Trans. Pattern Anal. Mach. Intell. 34 (5) (2012) 987–1002.
[118] A. Milan, K. Schindler, S. Roth, Detection- and trajectory-level exclusion in multiple object tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Portland, OR, USA, 2013, pp. 3682–3689.
[119] K.C.A. Kumar, C.D. Vleeschouwer, Discriminative label propagation for multi-object tracking with sporadic appearance features, in: Proc. IEEE Int. Conf. Comput. Vis., Sydney, Australia, 2013, pp. 2000–2007.
[120] W. Luo, B. Stenger, X. Zhao, T.-K. Kim, Automatic topic discovery for multi-object tracking, in: Proc. AAAI Conf. Artificial Intel., Austin, Texas, USA, 2015, pp. 1–8.
[121] S.-I. Yu, Y. Yang, A. Hauptmann, Harry potter's marauder's map: localizing and tracking multiple persons-of-interest by nonnegative discretization, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Portland, OR, USA, 2013, pp. 3714–3720.
[122] A. Andriyenko, K. Schindler, Multi-target tracking by continuous energy minimization, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Colorado Springs, CO, USA, 2011, pp. 1265–1272.
[123] P. Felzenszwalb, R. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Trans. Pattern Anal. Mach. Intell. 32 (9) (2010) 1627–1645.
[124] G. Shu, A. Dehghan, O. Oreifej, E. Hand, M. Shah, Part-based multiple-person tracking with partial occlusion handling, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Providence, RI, USA, 2012, pp. 1815–1821.
[125] K. Fragkiadaki, W. Zhang, G. Zhang, J. Shi, Two-granularity tracking: mediating trajectory and detection graphs for tracking under occlusions, in: Proc. Eur. Conf. Comput. Vis., Florence, Italy, 2012, pp. 552–565.
[126] S. Tang, M. Andriluka, B. Schiele, Detection and tracking of occluded people, Int. J. Comput. Vis. 110 (1) (2014) 58–69.
[127] S. Tang, M. Andriluka, A. Milan, K. Schindler, S. Roth, B. Schiele, Learning people detectors for tracking in crowded scenes, in: Proc. IEEE Int. Conf. Comput. Vis., Sydney, Australia, 2013, pp. 1049–1056.
[128] M.S. Ryoo, J.K. Aggarwal, Observe-and-explain: a new approach for multiple hypotheses tracking of humans and objects, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Anchorage, AL, USA, 2008, pp. 1–8.
[129] A. Andriyenko, S. Roth, K. Schindler, An analytical formulation of global occlusion reasoning for multi-target tracking, in: Proc. IEEE Int. Conf. Comput. Vis. Workshops, Barcelona, Spain, 2011, pp. 1839–1846.
[130] T.E. Fortmann, Y. Bar-Shalom, M. Scheffe, Sonar tracking of multiple targets using joint probabilistic data association, IEEE J. Ocean. Eng. 8 (3) (1983) 173–184.
[131] Y. Ban, S. Ba, X. Alameda-Pineda, R. Horaud, Tracking multiple persons based on a variational bayesian model, in: Proc. Eur. Conf. Comput. Vis., Workshops, Amsterdam, Netherlands, 2016, pp. 52–67.
[132] Y. Jin, F. Mokhtarian, Variational particle filter for multi-object tracking, in: Proc. IEEE Int. Conf. Comput. Vis., Rio de Janeiro, Brazil, 2007, pp. 1–8.
[133] C. Yang, R. Duraiswami, L. Davis, Fast multiple object tracking via a hierarchical particle filter, in: Proc. IEEE Int. Conf. Comput. Vis., vol. 1, Beijing, China, 2005, pp. 212–219.
[134] R. Hess, A. Fern, Discriminatively trained particle filters for complex multi-object tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Miami, FL, USA, 2009, pp. 240–247.
[135] B. Han, S.-W. Joo, L.S. Davis, Probabilistic fusion tracking using mixture kernel-based bayesian filtering, in: Proc. IEEE Int. Conf. Comput. Vis., Rio de Janeiro, Brazil, 2007, pp. 1–8.
[136] B. Wu, R. Nevatia, Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors, Int. J. Comput. Vis. 75 (2) (2007) 247–266.
[137] A.A. Perera, C. Srinivas, A. Hoogs, G. Brooksby, W. Hu, Multi-object tracking through simultaneous long occlusions and split-merge conditions, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 1, New York City, NY, USA, 2006, pp. 666–673.
[138] J.K. Wolf, A.M. Viterbi, G.S. Dixon, Finding the best set of k paths through a trellis with application to multitarget tracking, IEEE Trans. Aerosp. Electron. Syst. 25 (2) (1989) 287–296.
[139] H. Jiang, S. Fels, J.J. Little, A linear programming approach for multiple object tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Minneapolis, MN, USA, 2007, pp. 1–8.
[140] J. Berclaz, F. Fleuret, P. Fua, Multiple object tracking using flow linear programming, in: Proc. IEEE Int. Workshop Perform. Eval. Track. Surveillance, Snowbird, UT, 2009, pp. 1–8.
[141] A. Andriyenko, K. Schindler, Globally optimal multi-target tracking on a hexagonal lattice, in: Proc. Eur. Conf. Comput. Vis., Hersonissos, Greece, 2010, pp. 466–479.
[142] B. Leibe, K. Schindler, L. Van Gool, Coupled detection and trajectory estimation for multi-object tracking, in: Proc. IEEE Int. Conf. Comput. Vis., Rio de Janeiro, Brazil, 2007, pp. 1–8.
[143] Z. Wu, T.H. Kunz, M. Betke, Efficient track linking methods for track graphs using network-flow and set-cover techniques, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Colorado Springs, CO, USA, 2011, pp. 1185–1192.
[144] S. Tang, B. Andres, M. Andriluka, B. Schiele, Subgraph decomposition for multi-target tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Boston, MA, USA, 2015, pp. 5033–5041.
[145] S. Tang, B. Andres, M. Andriluka, B. Schiele, Multi-person tracking by multicut and deep matching, in: Proc. Eur. Conf. Comput. Vis., Workshops, Amsterdam, Netherlands, 2016, pp. 100–111.
[146] S. Tang, M. Andriluka, B. Andres, B. Schiele, Multiple people tracking by lifted multicut and person re-identification, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Honolulu, HI, USA, 2017, pp. 3539–3548.
[147] A. Dehghan, S. Modiri Assari, M. Shah, GMMCP tracker: globally optimal generalized maximum multi clique problem for multiple object tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Boston, MA, USA, 2015, pp. 4091–4099.
[148] P. Lenz, A. Geiger, R. Urtasun, FollowMe: efficient online min-cost flow tracking with bounded memory and computation, in: Proc. IEEE Int. Conf. Comput. Vis., Santiago, Chile, 2015, pp. 4364–4372.

[149] V. Chari, S. Lacoste-Julien, I. Laptev, J. Sivic, On pairwise costs for network flow multi-object tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Boston, MA, USA, 2015, pp. 5537–5545.
[150] A. Gaidon, Q. Wang, Y. Cabon, E. Vig, Virtual worlds as proxy for multi-object tracking analysis, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Las Vegas, NV, USA, 2016, pp. 4340–4349.
[151] N. Le, A. Heili, J.-M. Odobez, Long-term time-sensitive costs for CRF-based tracking by detection, in: Proc. Eur. Conf. Comput. Vis., Workshops, Amsterdam, Netherlands, 2016, pp. 43–51.
[152] J. Xu, Y. Cao, Z. Zhang, H. Hu, Spatial-temporal relation networks for multi-object tracking, in: Proc. IEEE Int. Conf. Comput. Vis., Seoul, Korea, 2019, pp. 3988–3998.
[153] P. Chu, H. Ling, FAMNet: joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking, in: Proc. IEEE Int. Conf. Comput. Vis., Seoul, Korea, 2019, pp. 6172–6181.
[154] I. Leichter, E. Krupka, Monotonicity and error type differentiability in performance measures for target detection and tracking in video, IEEE Trans. Pattern Anal. Mach. Intell. 35 (10) (2013) 2553–2560.
[155] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M. Boonstra, V. Korzhova, J. Zhang, Framework for performance evaluation of face, text, and vehicle detection and tracking in video: data, metrics, and protocol, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 319–336.
[156] B. Keni, S. Rainer, Evaluating multiple object tracking performance: the clear mot metrics, EURASIP J. Image Video Process. 2008 (2008) 1.
[157] B. Ristic, B.-N. Vo, D. Clark, B.-T. Vo, A metric for performance evaluation of multi-target tracking algorithms, IEEE Trans. Signal Process. 59 (7) (2011) 3452–3457.
[158] M. Kristan, J. Matas, A. Leonardis, T. Vojir, R. Pflugfelder, G. Fernandez, G. Nebehay, F. Porikli, L. Čehovin, A novel performance evaluation methodology for single-target trackers, IEEE Trans. Pattern Anal. Mach. Intell. 38 (11) (2016) 2137–2155.
[159] H. Possegger, T. Mauthner, P.M. Roth, H. Bischof, Occlusion geodesics for online multi-object tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Columbus, OH, USA, 2014, pp. 1306–1313.
[160] A. Bewley, Z. Ge, L. Ott, F. Ramos, B. Upcroft, Simple online and realtime tracking, in: Proc. IEEE Int. Conf. Image Process., Arizona, USA, 2016, pp. 3464–3468.
[161] C. Kim, F. Li, A. Ciptadi, J.M. Rehg, Multiple hypothesis tracking revisited, in: Proc. IEEE Int. Conf. Comput. Vis., Santiago, Chile, 2015, pp. 4696–4704.
[162] A.R. Zamir, A. Dehghan, M. Shah, GMCP-tracker: global multi-object tracking using generalized minimum clique graphs, in: Proc. Eur. Conf. Comput. Vis., Florence, Italy, 2012, pp. 343–356.
[163] S.-H. Bae, K.-J. Yoon, Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Columbus, OH, USA, 2014, pp. 1218–1225.
[164] H.S. Rezatofighi, A. Milan, Z. Zhang, Q. Shi, A. Dick, I. Reid, Joint probabilistic data association revisited, in: Proc. IEEE Int. Conf. Comput. Vis., Santiago, Chile, 2015, pp. 3047–3055.
[165] F. Solera, S. Calderara, R. Cucchiara, Learning to divide and conquer for online multi-target tracking, in: Proc. IEEE Int. Conf. Comput. Vis., Santiago, Chile, 2015, pp. 4373–4381.
[166] J. Yang, P.A. Vela, Z. Shi, J. Teizer, Probabilistic multiple people tracking through complex situations, in: Proc. IEEE Int. Workshop Perform. Eva. Track. Surveillance, Snowbird, Utah, USA, 2009, pp. 1–8.
[167] A. Alahi, L. Jacques, Y. Boursier, P. Vandergheynst, Sparsity-driven people localization algorithm: evaluation in crowded scenes environments, in: Proc. IEEE Int. Conf. Adv. Video Signal-based Surveillance, Genova, Italy, 2009, pp. 1–8.
[168] D. Conte, P. Foggia, G. Percannella, M. Vento, Performance evaluation of a people tracking system on PETS2009 database, in: Proc. IEEE Int. Conf. Adv. Video Signal-based Surveillance, Boston, MA, USA, 2010, pp. 119–126.
[169] L. Wen, W. Li, J. Yan, Z. Lei, D. Yi, S.Z. Li, Multiple target tracking based on undirected hierarchical relation hypergraph, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Columbus, OH, USA, 2014, pp. 1282–1289.
[170] H.B. Shitrit, J. Berclaz, F. Fleuret, P. Fua, Tracking multiple people under global appearance constraints, in: Proc. IEEE Int. Conf. Comput. Vis., Barcelona, Spain, 2011, pp. 137–144.
[171] L. Leal-Taixé, G. Pons-Moll, B. Rosenhahn, Everybody needs somebody: modeling social and grouping behavior on a linear programming multiple people tracker, in: Proc. IEEE Int. Conf. Comput. Vis. Workshops, Barcelona, Spain, 2011, pp. 120–127.
[172] M.D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, L. Van Gool, Online multiperson tracking-by-detection from a single, uncalibrated camera, IEEE Trans. Pattern Anal. Mach. Intell. 33 (9) (2011) 1820–1833.
[173] S. Zhang, J. Wang, Z. Wang, Y. Gong, Y. Liu, Multi-target tracking by learning local-to-global trajectory models, Pattern Recognit. 48 (2) (2015) 580–590.
[174] L. Leal-Taixé, G. Pons-Moll, B. Rosenhahn, Branch-and-price global optimization for multi-view multi-target tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Providence, RI, USA, 2012, pp. 1987–1994.
[175] A.V. Segal, I. Reid, Latent data association: Bayesian model selection for multi-target tracking, in: Proc. IEEE Int. Conf. Comput. Vis., Sydney, Australia, 2013, pp. 2904–2911.
[176] A. Dehghan, Y. Tian, P.H. Torr, M. Shah, Target identity-aware network flow for online multiple target tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Boston, MA, USA, 2015, pp. 1146–1154.
[177] M. Hofmann, D. Wolf, G. Rigoll, Hypergraphs for joint multi-view reconstruction and multi-object tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Portland, OR, USA, 2013, pp. 3650–3657.
[178] M. Hofmann, M. Haag, G. Rigoll, Unified hierarchical multi-object tracking using global data association, in: Proc. IEEE Int. Conf. Adv. Video Signal-based Surveillance, Krakow, Poland, 2013, pp. 22–28.
[179] X. Shi, H. Ling, W. Hu, C. Yuan, J. Xing, Multi-target tracking with motion context in tensor power iteration, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Columbus, OH, USA, 2014, pp. 3518–3525.
[180] Z. Wu, J. Zhang, M. Betke, Online motion agreement tracking, in: Proc. British Mach. Vis. Conf., Bristol, UK, 2013, p. 7.
[181] A. Geiger, P. Lenz, R. Urtasun, Are we ready for autonomous driving? The KITTI vision benchmark suite, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Providence, RI, USA, 2012, pp. 3354–3361.
[182] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, K. Schindler, MOT16: a benchmark for multi-object tracking, arXiv:1603.00831.
[183] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Columbus, OH, USA, 2014, pp. 580–587.
[184] R. Girshick, Fast R-CNN, in: Proc. IEEE Int. Conf. Comput. Vis., Santiago, Chile, 2015, pp. 1440–1448.
[185] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, in: Proc. Neural Info. Process. Sys., Montreal, Canada, 2015, pp. 91–99.
[186] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Las Vegas, NV, USA, 2016, pp. 779–788.
[187] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, SSD: single shot multibox detector, in: Proc. Eur. Conf. Comput. Vis., Amsterdam, Netherlands, 2016, pp. 21–37.
[188] H. Hu, J. Gu, Z. Zhang, J. Dai, Y. Wei, Relation networks for object detection, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Salt Lake City, UT, USA, 2018, pp. 3588–3597.

[189] J. Dai, Y. Li, K. He, J. Sun, R-FCN: object detection via region-based fully convolutional networks, in: Proc. Neural Info. Process. Sys., Barcelona, Spain, 2016, pp. 379–387.
[190] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, Y. Wei, Deformable convolutional networks, in: Proc. IEEE Int. Conf. Comput. Vis., Venice, Italy, 2017, pp. 764–773.
[191] G. Shu, A. Dehghan, M. Shah, Improving an object detector and extracting regions using superpixels, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Portland, OR, USA, 2013, pp. 3721–3727.
[192] E. Ristani, F. Solera, R. Zou, R. Cucchiara, C. Tomasi, Performance measures and a data set for multi-target, multi-camera tracking, in: Proc. Eur. Conf. Comput. Vis., Workshops, Amsterdam, Netherlands, 2016, pp. 17–35.
[193] E. Ristani, C. Tomasi, Features for multi-target multi-camera tracking and re-identification, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Salt Lake City, UT, USA, 2018, pp. 6036–6046.
[194] Y. Park, V. Lepetit, W. Woo, Multiple 3D object tracking for augmented reality, in: Proc. IEEE/ACM Int. Symp. Mix. Augment, Real., Cambridge, UK, 2008, pp. 117–120.
[195] B. Zhou, X. Wang, X. Tang, Understanding collective crowd behaviors: learning a mixture model of dynamic pedestrian-agents, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Providence, RI, USA, 2012, pp. 2871–2878.
[196] B. Zhou, X. Tang, X. Wang, Coherent filtering: detecting coherent motions from crowd clutters, in: Proc. Eur. Conf. Comput. Vis., Florence, Italy, 2012, pp. 857–871.
[197] A. Krizhevsky, I. Sutskever, G. Hinton, Imagenet classification with deep convolutional neural networks, in: Proc. Neural Info. Process. Sys., Lake Tahoe, USA, 2012, pp. 1097–1105.
[198] F. Yu, W. Li, Q. Li, Y. Liu, X. Shi, J. Yan, POI: multiple object tracking with high performance detection and appearance feature, in: Proc. Eur. Conf. Comput. Vis., Workshops, Amsterdam, Netherlands, 2016, pp. 36–42.
[199] B. Lee, E. Erdenee, S. Jin, M.Y. Nam, Y.G. Jung, P.K. Rhee, Multi-class multi-object tracking using changing point detection, in: Proc. Eur. Conf. Comput. Vis., Workshops, Amsterdam, Netherlands, 2016, pp. 68–83.
[200] J. Son, M. Baek, M. Cho, B. Han, Multi-object tracking with quadruplet convolutional neural networks, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Honolulu, HI, USA, 2017, pp. 5620–5629.
[201] S. Schulter, P. Vernaza, W. Choi, M. Chandraker, Deep network flow for multi-object tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Honolulu, HI, USA, 2017, pp. 6951–6960.
[202] A. Sadeghian, A. Alahi, S. Savarese, Tracking the untrackable: learning to track multiple cues with long-term dependencies, in: Proc. IEEE Int. Conf. Comput. Vis., Venice, Italy, 2017, pp. 300–311.
[203] Q. Chu, W. Ouyang, H. Li, X. Wang, B. Liu, N. Yu, Online multi-object tracking using CNN-based single object tracker with spatial-temporal attention mechanism, in: Proc. IEEE Int. Conf. Comput. Vis., Venice, Italy, 2017, pp. 4836–4845.
[204] J. Zhu, H. Yang, N. Liu, M. Kim, W. Zhang, M.-H. Yang, Online multi-object tracking with dual matching attention networks, in: Proc. Eur. Conf. Comput. Vis., Munich, German, 2018, pp. 366–382.
[205] L. Wang, W. Ouyang, X. Wang, H. Lu, Visual tracking with fully convolutional networks, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Boston, MA, USA, 2015, pp. 3119–3127.
[206] A. Milan, L. Leal-Taixe, K. Schindler, I. Reid, Joint tracking and segmentation of multiple targets, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Boston, MA, USA, 2015, pp. 5397–5406.
[207] Y. Jun Koh, C.-S. Kim, CDTS: collaborative detection, tracking, and segmentation for online multiple object segmentation in videos, in: Proc. IEEE Int. Conf. Comput. Vis., Venice, Italy, 2017, pp. 3601–3609.
[208] P. Voigtlaender, M. Krause, A. Osep, J. Luiten, B.B.G. Sekar, A. Geiger, B. Leibe, MOTS: multi-object tracking and segmentation, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Long Beach, CA, USA, 2019, pp. 7942–7951.
[209] C. Su, S. Zhang, J. Xing, W. Gao, Q. Tian, Deep attributes driven multi-camera person re-identification, in: Proc. Eur. Conf. Comput. Vis., Amsterdam, Netherlands, 2016, pp. 475–491.
[210] Z. Tang, M. Naphade, M.-Y. Liu, X. Yang, S. Birchfield, S. Wang, R. Kumar, D. Anastasiu, J.-N. Hwang, CityFlow: a city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Long Beach, CA, USA, 2019, pp. 8797–8806.
[211] E. Insafutdinov, M. Andriluka, L. Pishchulin, S. Tang, E. Levinkov, B. Andres, B. Schiele, ArtTrack: articulated multi-person tracking in the wild, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Honolulu, HI, USA, 2017, pp. 6457–6465.
[212] U. Iqbal, A. Milan, M. Andriluka, E. Ensafutdinov, L. Pishchulin, J. Gall, S. B., PoseTrack: a benchmark for human pose estimation and tracking, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Salt Lake City, UT, USA, 2018, pp. 5167–5176.
[213] S. Jin, W. Liu, W. Ouyang, C. Qian, Multi-person articulated tracking with spatial and temporal embeddings, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Long Beach, CA, USA, 2019, pp. 5664–5673.
[214] Y. Raaj, H. Idrees, G. Hidalgo, Y. Sheikh, Efficient online multi-person 2D pose tracking with recurrent spatio-temporal affinity fields, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Long Beach, CA, USA, 2019, pp. 4620–4628.