Recent Advances in Embedding Methods for Multi-Object Tracking: A Survey

Gaoang Wang, *Member, IEEE,* Mingli Song, *Senior Member, IEEE,* and Jenq-Neng Hwang, *Fellow, IEEE*

Abstract—Multi-object tracking (MOT) aims to associate target objects across video frames in order to obtain entire moving trajectories. With the advancement of deep neural networks and the increasing demand for intelligent video analysis, MOT has gained significantly increased interest in the computer vision community. Embedding methods play an essential role in object location estimation and temporal identity association in MOT. Unlike other computer vision tasks, such as image classification, object detection, re-identification, and segmentation, embedding methods in MOT have large variations, and they have never been systematically analyzed and summarized. In this survey, we first conduct a comprehensive overview with in-depth analysis for embedding methods in MOT from seven different perspectives, including patch-level embedding, single-frame embedding, cross-frame joint embedding, correlation embedding, sequential embedding, tracklet embedding, and cross-track relational embedding. We further summarize the existing widely used MOT datasets and analyze the advantages of existing state-of-the-art methods according to their embedding strategies. Finally, some critical yet under-investigated areas and future research directions are discussed.

Index Terms—Multi-Object Tracking, Embedding Methods, Literature Survey, Evaluation Metric, State-of-the-Art Analysis.

1 Introduction

Multi-object tracking (MOT) has been widely studied in recent years, aiming to associate detected objects across video frames to obtain the entire moving trajectories. Recent years have seen the emergence of a variety of tracking algorithms, from graph clustering methods [1], [2], [3], [4] to graph neural networks [5], [6], [7], [8] that aggregate information across frames and objects; from tracking-by-detection paradigm [9], [10], [11] to joint detection and tracking [5], [12], [13], [14], [15], [16] to improve the detection performance with multiple frames; from Kalman filtering [17] to recurrent neural networks (RNN) [18] and long-short term memory (LSTM) [19] to boost association performance with the motion clue. With the development of tracking algorithms, MOT can be applied in many tasks, such as traffic flow analysis [1], [20], [21], [22], human behavior prediction and pose estimation [23], [24], [25], [26], autonomous driving assistance [27], [28], and even for underwater animal abundance estimation [29], [30], [31].

The flow of MOT system can be mainly divided into two parts, *i.e.*, embedding model and association algorithm. With input multiple successive frames, the object locations and track identities (IDs) are estimated via the embedding techniques and association approaches. MOT is challenging due to the presence of illumination changes, occlusions, complex environments, fast camera movement, unreliable detections, varying low-image resolutions [32], etc. In addition, the tracking performance can be affected by individual steps of tracking algorithms, such as detection, feature extraction, affinity estimation, and association. These result in significant variations and uncertainty. With the recent progress on representation learning with deep neural networks, embedding methods play an essential role in object location estimation and temporal identity association in MOT. In this survey, we focus more on the review of embedding learning, rather than association, though association is also important in MOT.

However, the embedding learning approaches in MOT have never been systematically analyzed and summarized. Unlike other computer vision tasks, such as image classification, object detection, re-identification (Re-ID), and segmentation, embedding methods in MOT have large variations. Some embedding methods combine multiple-task heads [16], [33], [34], [35], [36], including box regression, object classification, and re-identification. Some embedding methods consider spatial-temporal correlations [12], [14], [37], [38], [39], collaborating both appearance and motion information. Some methods exploit the interaction relationships among objects, foreground and background, local and global information with correlation and attention, to learn the track embeddings [40], [41], [42], [43]. The large deviation of embedding methods motivates us to conduct a comprehensive survey from an embedding perspective and discuss several under-investigated embedding areas and future directions.

There are some existing surveys [44], [45], [46], [47], [48] of MOT published in recent years. Specifically, [44] summarizes some deep learning based trackers and deep neural network structures. [45] focuses on the review of model-based multiple hypotheses tracking with machine learning techniques in detection, filtering, and association. [46] provides a review of how deep learning is adopted in MOT, including detection, feature extraction, affinity com-

- G. Wang is with the Zhejiang University-University of Illinois at Urbana-Champaign Institute and the College of Computer Science and Technology, Zhejiang University, Zhejiang, China.
  E-mail: gaoangwang@intl.zju.edu.cn
- M. Song is with the College of Computer Science and Technology, Zhejiang University, Zhejiang, China.
  E-mail: brooksong@zju.edu.cn
- J.-N. Hwang is with the Department of Electrical and Computer Engineering, the University of Washington, WA, USA.
  E-mail: hwang@uw.edu

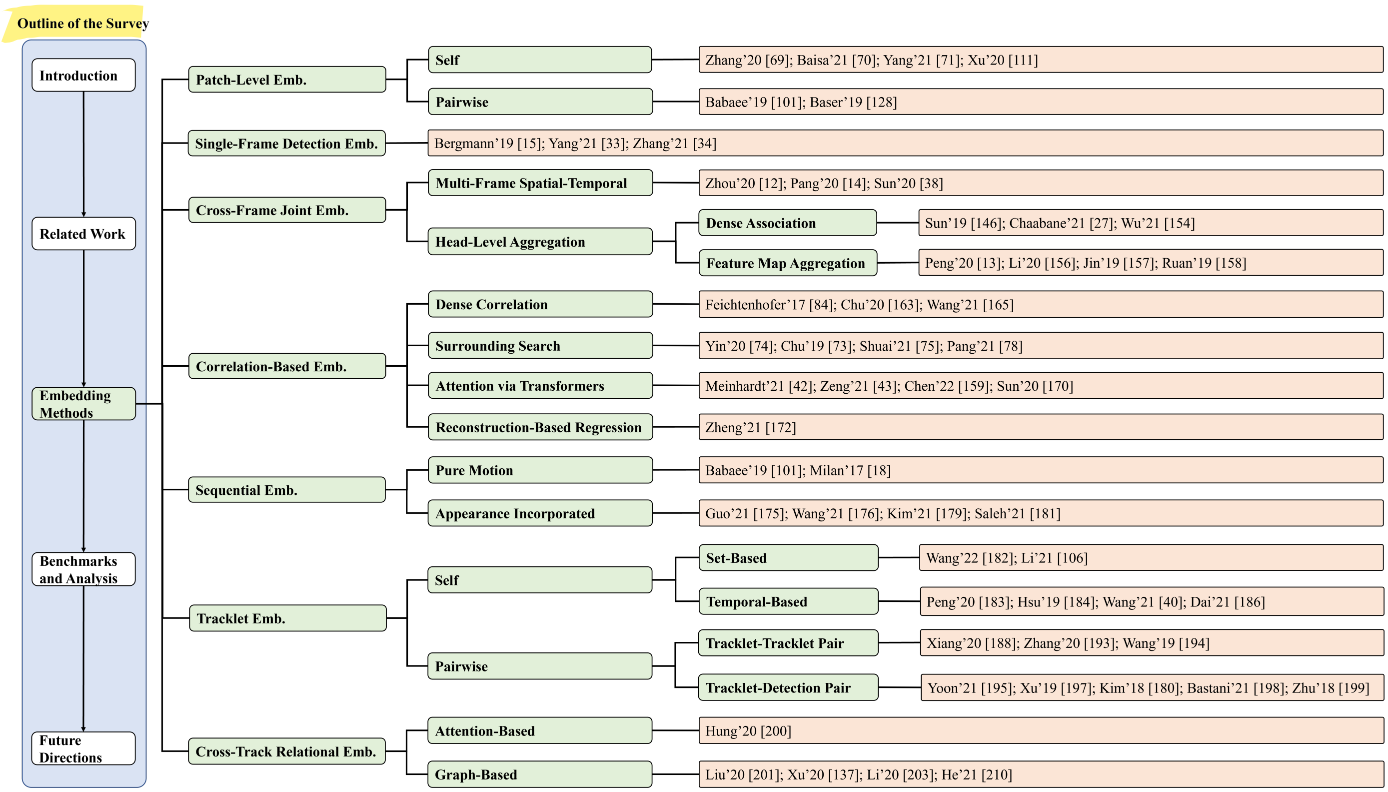

Fig. 1. A taxonomy of embedding methods in MOT. The top flowchart is the outline of this survey. The green and light red boxes show the embedding methods and representative references, respectively.

putation, and association. [47] reviews the evolution of MOT in recent decades, focusing on deep learning techniques and investigating the recent advances in MOT. [48] provides a review of the MOT system and discusses approaches from different aspects. Unlike all existing surveys, we focus on embedding learning in MOT, *i.e.*, how to learn object oriented representative features for the MOT task, and comprehensively analyze state-of-the-art methods according to the embedding strategies. The main contributions of this survey are summarized as follows:

- We categorize and summarize existing embedding methods designed for MOT and make an in-depth and comprehensive analysis by discussing the advantages and limitations of the methods. The summary provides insights for future algorithm design and new topic exploration.
- We summarize the widely used datasets and benchmarks and analyze the state-of-the-art approaches according to the embedding methods.
- We attempt to discuss several important research directions with under-investigated issues related to embedding techniques and take a step towards future trends in MOT.

The outline of this survey is summarized as follows. We first demonstrate related works, including the most relevant tasks of MOT in Section 2. A taxonomy and detailed review of embedding methods are provided in Section 3. We then summarize the existing widely used MOT datasets, evaluation metrics and analyze the state-of-the-art approaches according to the embedding methods in Section 4. We discuss several under-investigated issues and point out the trends and potential future study in Section 5. Conclusions is drawn in Section 6.

2 MOT Related Tasks

Three tasks, *i.e.*, single object tracking (SOT), video object detection (VOD), and re-identification (Re-ID), are highly related to MOT. Many embedding methods in the MOT field are inspired by these tasks. A comparison and brief review of these tasks are demonstrated in this section.

2.1 Single Object Tracking

Single-object tracking (SOT), also named visual object tracking (VOT), aims to estimate an unknown visual target trajectory when only an initial state of the target (in a video frame) is available [49]. The tracking target is purely determined by the first frame and does not rely on any categories. Inspired by deep learning breakthroughs [50], [51], [52], [53], [54] in ImageNet large-scale visual recognition competition (ILSVRC) [55] and visual object tracking (VOT) challenge [56], [57], embedding learning methods have attracted considerable interest in the visual tracking community to provide robust trackers. Many works attempt the tracking by learning discriminative target representations, such as learning distractor-aware [58] or target-aware features [59], leveraging different types of deep features such as context information [60], [61] or temporal features/models [62], [63], full exploring of low-level spatial features [64], [65], employing correlation-guided attention modules to exploit the relationship between the template and RoI feature maps [66], and computing correlations between attentional features [67].

Unlike SOT, initial states of objects are unknown in the MOT task, requiring pre-defined categories for tracking. As a result, MOT methods either adopt off-the-shelf detection following a tracking-by-detection scheme [68], [69], [70], [71] or exploit the joint detection and tracking models [12], [15], [33], [34], [37]. Some works incorporate embedding methods from SOT into MOT with correlation and attention techniques [72], [73], [74], [75], [76], [77], [78].

2.2 Video Object Detection

Video object detection (VOD) aims to detect objects across multiple video frames by jointly recognizing objects and estimating locations [79]. Embedding methods are also important in the VOD task. Embedding learning can filter the features of a video frame, select relatively representative features, propagate key features for detection, and delineate the key areas that deserve feature filtering in subsequent frames. For example, [80] proposes a special attention network known as the progressive sparse local attention (PSLA) framework to propagate features between different video frames. STMN [81] uses a module that calculates similar correlations to locally align the feature map. [82] introduces Patchwork, a method that uses the attention mechanism to predict the position of an object in the next frame to solve the video object detection problem. [83] proposes a spatial-temporal sampling network (STSN) that introduces deformable convolution to detect video frames.

Some embedding methods in VOD can be directly applied for MOT, such as [84]. However, unlike VOD, MOT usually does not require the recognition of objects unless objects from multiple categories are tracked jointly. Besides, track ID prediction is essential in the MOT task but not required in VOD. As a result, the embedding methods in MOT should be discriminative among objects even in the same category.

2.3 Re-Identification

Re-identification (Re-ID) aims at verifying object identity from different collections of images, usually from varying angles, illumination, and poses in different cameras [85], [86], [87]. With the advancement of deep neural networks and increasing demand for intelligent video surveillance, Re-ID has significantly increased interest in the computer vision community. Much progress has been made for feature representation learning in Re-ID. Such embedding methods include global feature learning that extracts global feature representation for each image without additional annotation cues [88], local feature learning that aggregates part-level local features to formulate a combined representation [89], [90], [91], auxiliary feature learning that improves the feature representation learning using auxiliary information like attributes [92], [93], [94], and video feature learning that learns video representation for video-based Re-ID [95] using multiple image frames and temporal information [96], [97].

Inspired by the Re-ID task, many MOT works employ Re-ID based embedding methods in the MOT task to learn the object appearance embeddings [70], [72], [98]. Unlike the Re-ID task that does not require temporal information between query and gallery images, temporal consistency is a more important cue in MOT.

TABLE 1
Notations used in the Survey

Symbol	Description
$\boldsymbol{X}^t$	Input image of t-th frame.
$\boldsymbol{F}^t$	Feature maps of t-th frame.
$\boldsymbol{D}_i^t$	i-th detection at t-th frame, containing cropped image and box information.
$\mathcal{T}_i$	Tracklet containing sequential detections, *i.e.*, $\mathcal{T}_i = [\boldsymbol{D}_i^{t-\tau}; \boldsymbol{D}_i^{t-\tau+1}; ...; \boldsymbol{D}_i^t]$.
$\boldsymbol{z}_i^t$	Embedding of i-th detection at t-th frame.
$\boldsymbol{z}_{i,j}$	Joint embedding of i-th and j-th detections or tracklets.
$f(\cdot), g(\cdot)$	Embedding networks.

3 A Taxonomy of Embedding Methods in MOT

3.1 Overview and Notation

Embedding methods are essential for object location estimation and ID association. Our proposed taxonomy of MOT embedding methods is shown in Fig. 1. In this section, we categorize the commonly used MOT embedding methods into seven groups, including patch-level embedding, single-frame embedding, cross-frame joint embedding, correlation-based embedding, sequential embedding, tracklet embedding, and cross-track relational embedding. For each category, we introduce the representative algorithms and then discuss the weaknesses and strengths, hoping to provide a thorough analysis of each type of method to researchers. For better demonstration, we first summarize the notations used in this survey in Table 1.

3.2 Patch-Level Box Image Embedding

Patch-level box image embedding is essential for the tracking-by-detection scheme. Some approaches treat MOT as a Re-ID problem. Most of the existing methods in this category focus on embedding individual detections [68], [69], [70], [71], while a few methods try to model the relation of two detected objects directly with pairwise embedding strategies [99], [100], [101]. As a result, we categorize the patch-level box image embedding into self-embedding and pairwise embedding subgroups.

3.2.1 *Self-Embedding*

Given the cropped box image based on the detection $\boldsymbol{D}_i$, the self-embedding represents the appearance feature of the detected object. This can be formulated as follows,

$$\boldsymbol{z}_i = f(\boldsymbol{D}_i), \tag{1}$$

where $f(\cdot)$ represents the embedding network, $\boldsymbol{z}_i$ represents the embedding of detection $\boldsymbol{D}_i$. To learn discriminative features to distinguish different object identities, the Re-ID framework is widely used for the embedding.

Some existing approaches treat the same objects across different frames as individual classes and employ cross-entropy loss for ID classification to learn the embedding of the cropped detection images, such as [68], [69], [70], [72], [102]. Some methods adopt triplet loss [71], [103], [104], [105] and softmax-based contrastive loss [106] to learn discriminative embeddings within batch samples, in which detections from the same object are treated as positive samples and that from different objects are treated as negative samples. For the triplet loss, tuples of anchor, positive, and negative samples are constructed and exploited for distance metric learning, with the same strategy used in FaceNet [107]. Compared with cross-entropy loss, triplet loss can better distinguish the subtle difference among different IDs, especially for objects with similar color and poses. For the softmax-based contrastive loss, maximizing the similarity of positive pairs [108] is adopted. Unlike triplet loss that only takes three samples each time, contrastive loss compare the relations of all negative pairs in the batch. Combining multiple losses in the embedding learning can also improve the tracking performance [109].

Except for the bounding boxes, some works take advantage of segmentation masks for learning embeddings. Segmentation-based MOT can improve the effectiveness of the embedding, especially for occluded objects, and alleviate the influence from the background regions in the bounding box. For example, based on the instance segmentation results [110], PointTrack [111], [112] treats the pixels of the cropped image as a 2D point cloud, and the points from the foreground target and background environment are jointly learned for each cropped detection image.

For embedding networks, ResNet-50 [54] is widely adopted in many works, such as [68], [70], [71], [105], [109]. Besides that, some works [69], [72], [98], [103], [113] adopt wide residual network [114], GoogLeNet [53], PCB network [91], and even customized networks [102].

To learn embedding that can be generalized to diverse scenarios, most approaches use additional datasets from the person Re-ID task for training. Commonly used Re-ID datasets include Market-1501 [115], CUHK01 and CUHK03 [116], DukeMTMC [117], VIPeR [118], [119], i-LIDS [119], and MSMT17 [120]. Some approaches use baseline trackers like SORT [121] to generate object IDs based on unlabeled video datasets, such as AVA-Kinetics [122], [123] and PathTrack [124].

There are several benefits for the patch-level self-embedding type of method. First, it is more flexible with variant detection algorithms, Re-ID models, and association methods, since each part can be trained separately. As a result, it can always combine off-the-shelf state-of-the-art detectors and Re-ID models. Second, since it takes cropped images as individual samples, it can employ a large amount of data in training. Compared with fully annotated video data, large-scale cropped image datasets are easier to collect. On the other side, the weakness is also obvious. It takes multiple stages in training, resulting in sub-optimal solutions. Besides, temporal information is hard to be learned in the embedding with cropped image samples.

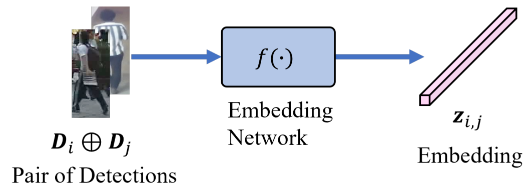

Fig. 2. Pairwise patch-level embedding. $\oplus$ represents the concatenation.

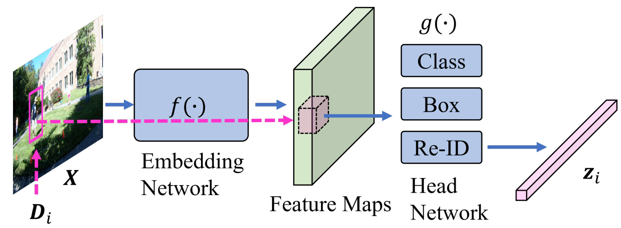

Fig. 3. Single-frame joint detection embedding.

3.2.2 Pairwise Embedding

Rather than learning embeddings from individual detections, pairwise embedding networks take pairs of detections and directly learn the similarity between two detected objects. A binary classifier is usually adopted to indicate whether the two detections belong to the same target, formulated as follows,

$$z_{i,j} = f(D_i, D_j), \tag{2}$$

where $z_{i,j}$ is the pairwise embedding of two detections D_i and D_j.

For example, some works [99], [100], [101] concatenate a pair of cropped images as input and use binary cross-entropy loss to distinguish whether two detections belong to the same object. Some works [125] concatenates intermediate feature maps for embedding. In addition, some works also take motion cues into consideration. Specifically, [126] concatenates both cropped image pairs and the corresponding optical flow maps [127] for estimation. [128] includes appearance branch as well as motion branch with 3D box parameters in the siamese network, trained with cosine similarity as a binary classification problem.

For pairwise embedding, it has similar strengths as patch-level self-embedding. With paired input, the discrimination is easier to be learned among different IDs. However, the computation complex has been raised from $\mathcal{O}(n)$ to $\mathcal{O}(n^2)$, where n denotes the number of cropped detections.

3.3 Single-Frame Detection Embedding

Frame-based embedding jointly learns detection and Re-ID features in an end-to-end manner. Given an input frame X, the network learns the discriminative feature of each detection as follows,

$$\{z_i | i = \{1, 2, ..., |\mathcal{D}|\}\} = f(X), \tag{3}$$

where $|\mathcal{D}|$ is the number of detections of the input frame, z_i is the embedding for the corresponding detection D_i.

For single-frame based embedding, existing works usually follow the state-of-the-art detector frameworks. Some works use anchor-based frameworks for embedding learning. Specifically, [15], [33] adopt Faster R-CNN [129]; [36] follows RetinaNet [130]; and [16] exploits the YOLOv3 framework with DarkNet-53 [131]. However, as mentioned in [34], anchors are not fit for Re-ID features in nature. As a result, some works adopt anchor-free frameworks. For example, [34], [132] build upon CenterNet [133] with deep layer aggregation (DLA) [12], [134], [135] used in the backbone architecture.

One of the major challenges for jointly learning embeddings for detection and Re-ID arises from the conflict of the two tasks. The detection task aims to identify object categories, such as pedestrian and vehicle, from the background, while Re-ID embeddings aim to distinguish distinct objects rather than the class. Some works decouple the embeddings for different tasks to address such an issue for multi-task learning. Specifically, [33], [34] design a separate Re-ID head to learn discriminative embeddings in addition to the original detection branch. [35] decouples detection and Re-ID with the proposed global context disentangling and uses guided Transformer encoder with deformable attention to extract Re-ID features. [36] designs task-specific post-feature pyramid network (FPN) layers for classification, bounding box regression, and generates embedding vectors simultaneously. Similarly, [16] uses FPN for task-specific embedding. [136] proposes a CCPNet that decouples different tasks with multiple decoders to learn instance segmentation and ID-based embeddings for multi-object tracking and segmentation (MOTS) tasks. [132] designs an unsupervised Re-ID framework based on strong and weak supervision of the the association cues from two frames following FairMOT [34]. Moreover, [33] employs the differentiable multi-objective tracking metric dMOTA and dMOTP [137] for training.

For single-frame based embedding, detection accuracy has a large effect on the tracking accuracy. As a result, some detection datasets, such as COCO [138], ETH [139], CityPerson [140] and CrowdHuman [141], are widely used for detection branch. For datasets like CalTech [142], MOT17 [143], MOT20 [144], CUHK-SYSU [145] and PRW [88] that have both box and identity annotations, are often used to jointly learn the detection and Re-ID features. Compared with patch-level embedding, single-frame embedding methods do not rely on pre-trained detectors and do not require extra computation and storage to crop the detection patches. This one-stage training strategy increases the embedding efficiency. However, frame-based methods cannot efficiently utilize crop-based Re-ID datasets for training, causing a degradation for generalization. Besides, temporal consistency is still not well-exploited in the embedding framework.

3.4 Cross-Frame Joint Embedding

To jointly learn the appearance and temporal features across multiple frames, cross-frame embedding plays an important role in MOT. The embedding can be formulated as follows,

$$\{\boldsymbol{z}_i^t | i = \{1, 2, ..., |\mathcal{D}^t|\}\} = f([\boldsymbol{X}^{t-\tau}; \boldsymbol{X}^{t-\tau+1}; ...; \boldsymbol{X}^t]), \quad (4)$$

where $[\boldsymbol{X}^{t-\tau}; \boldsymbol{X}^{t-\tau+1}; ...; \boldsymbol{X}^t]$ represents the concatenation of multiple frames from $t - \tau$ to t. Some approaches [12], [14], [38] adopt embedding networks, such as 3D convolution and convolutional LSTMs, that learn spatial-temporal feature maps for tracking, and some approaches [13], [146], [147] extract features for individual frames and then aggregate embeddings to model temporal relations in the task-specific heads. These two sub-categories are demonstrated in the following sub-sections.

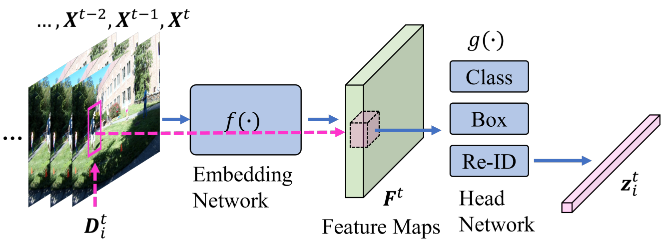

Fig. 4. Multi-frame spatial-temporal embedding.

3.4.1 Multi-Frame Spatial-Temporal Embedding

The multi-frame spatial-temporal embedding can be formulated as follows,

$$\begin{aligned} \boldsymbol{F}^t &= f([\boldsymbol{X}^{t-\tau}; \boldsymbol{X}^{t-\tau+1}; ...; \boldsymbol{X}^t]), \\ \{\boldsymbol{z}_i^t | i &= \{1, 2, ..., |\mathcal{D}^t|\}\} = g(\boldsymbol{F}^t), \end{aligned} \quad (5)$$

where $\boldsymbol{F}^t$ is the intermediate feature maps obtained from the backbone embedding network $f(\cdot)$ that learns spatial-temporal information from sequential frames $[\boldsymbol{X}^{t-\tau}; \boldsymbol{X}^{t-\tau+1}; ...; \boldsymbol{X}^t]$, and $g(\cdot)$ is a head network that generates the final embeddings $\{\boldsymbol{z}_i^t | i = \{1, 2, ..., |\mathcal{D}^t|\}\}$ of detections and tracklets.

Specifically, [14] takes 3D-ResNet [54], [148] as the backbone to generate object tubes and combines GIOU [149], focal loss [130], and binary cross-entropy loss in training. Similarly, DMM-Net [38] employs 3D convolution to learn spatial-temporal embeddings given multiple frames to generate tubes and predicts multi-frame motion, classes, and visibility to generate tracklets. Track R-CNN [37] follows the framework of Mask R-CNN [150] and aggregates the feature maps from multiple frames using 3D convolution and convolutional LSTM [151], [152] for temporal context embedding, which is an end-to-end method of segmentation based embedding for prediction and association. CenterTrack [12] follows the CenterNet framework [133] and concatenates a pair of sequential frames and the heat map of the previous frame for joint embedding, object center location estimation, as well as size and offset prediction. [39] takes multiple frames using an encoder-decoder architecture with temporal priors embedding based on short connections [153] to estimate multi-channel trajectory maps simultaneously, including presence map, appearance map, and motion map.

Thanks to the capability of learning temporal consistency with 3D neural networks and LSTMs, motion features can be incorporated into the embedding framework. On the other hand, it also increases the computational cost for training and testing. Current spatial-temporal embedding usually only considers a few frames for joint embedding. As a result,

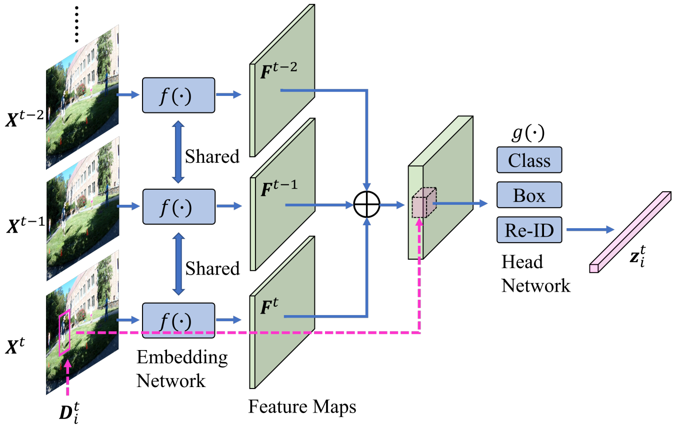

Fig. 5. Head-level aggregated embedding.

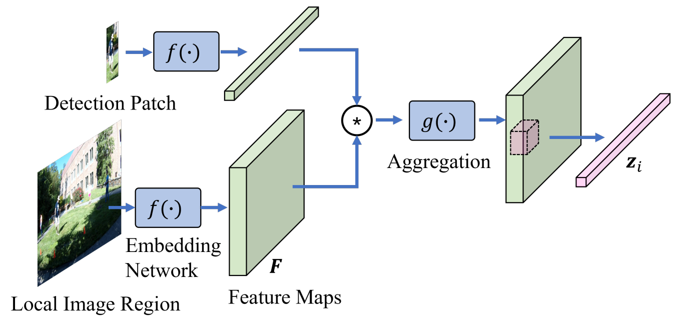

Fig. 6. Correlation-based embedding.

the learned temporal motion features are not robust enough to model the diverse movement of objects. Learning the long-time dependency also needs to be further studied in future works.

3.4.2 Head-Level Feature Aggregated Embedding

Slightly different from the multi-frame spatial-temporal embedding, methods of this sub-category usually extract features of input frames independently and then aggregate the feature maps in the head network for embedding, which can be formulated as follows,

$$\begin{aligned} &\boldsymbol{F}^t = f(\boldsymbol{X}^t), \\ &\{\boldsymbol{z}_i^t | i = \{1, 2, ..., |\mathcal{D}^t|\}\} = g([\boldsymbol{F}^{t-\tau}; \boldsymbol{F}^{t-\tau+1}; ...; \boldsymbol{F}^t]), \end{aligned} \tag{6}$$

where $f(\cdot)$ and $g(\cdot)$ represents the backbone network and head network, respectively.

Dense association. Some works aggregate embeddings for the dense association. Specifically, DAN [146] proposes a deep affinity network that predicts the dense association of detection center locations between a pair of frames using the features extracted from different layers of VGG Net [52]. Similar to DAN [146], DEFT [27] designs a matching head to aggregate the embeddings from pairs of frames. Besides that, motion forecasting with LSTM is exploited in the matching head for association in [27]. TraDeS [154] also jointly learns the offset, 2D, 3D, and mask estimation based on multiple frames dense embedding aggregation using cost volume-based association and deformable convolution [155].

Feature map aggregation. Apart from aggregating dense embeddings, some works concatenate feature maps from a pair of frames for prediction. For example, Chained-tracker [13] adopts the stacked feature maps for bounding box regression and ID verification following Faster R-CNN framework [129]. [156] concatenates feature maps from adjacent frames for paired 2D box regression and initial 3D box estimation based on spatial-temporal optimization. [157] learns a temporalNet to aggregate features from two frames with a combination of keypoint embedding branch and temporal instance embedding branch for both pose estimation and tracking. [158] proposes an end-to-end pose-guided insight network for the data association in multi-person pose tracking, which jointly learns feature extraction, similarity estimation, and identity assignment. PatchTrack [159] takes a pair of sequential feature maps to the Transformer encoder and uses the previously predicted tracks from Kalman Filtering [160] as queries in the Transformer decoder. TransCenter [147] also adopts Transformer for embedding in MOT, where dense pixel-level multi-scale detection and tracking queries are fed forward to two query learning networks based on deformable transformer encoder and decoder to obtain detection and tracking features. To learn temporal information, previous center heatmaps are also concatenated in the tracking branch.

Other suitable strategies are also exploited in the embedding aggregation. For example, [5] aggregates node embeddings based on graph neural networks (GNN) using GraphConv [161] given a pair of sequential frames. [162] generates fused targets based on proposed FuseTrack from detected objects and tracked targets with the guide of optical flows estimated from input frames.

Compared with multi-frame spatial-temporal embedding, head-level aggregated embedding methods encode each frame individually with shared backbones, largely reducing the computation cost. However, it may lack low-level pixel-wise correlation features between frames for detection and association.

3.5 Correlation-Based Embedding

Inspired by the SOT methods, target location can be refined via the correlation between detections and generated feature maps. The correlation-based approaches can be represented as follows,

$$\boldsymbol{z}_i = g(\boldsymbol{F} * f(\boldsymbol{D}_i)), \tag{7}$$

where $*$ denotes the correlation operation, $g(\cdot)$ and $f(\cdot)$ denotes the embedding networks for final embedding and detection embedding, respectively.

Dense correlation. Some works estimate the dense correlation feature maps. For example, [84] uses a correlation layer that learns the dense temporal relations given sequential feature maps. DASOT [163] integrates data association and SOT in a unified framework, in which dense correlation feature maps are estimated for the temporal association, built upon truncated ResNet-50 with feature pyramid network (FPN) [164]. In addition, [165] estimates both temporal correlation and multi-scale spatial correlation with dense feature maps.

Surrounding search. Some works conduct correlations between individual detections and surrounding local regions via SOT algorithms. Specifically, following deep tracker SiamFC [166], [74] uses the siamese network to calculate the correlation between anchor samples with both positive and negative local regions. Built upon lightweight AlexNet [50], the network adopts triplet loss for discriminative learning to generate embeddings. Following the Siamese-RPN tracker [167], [72] searches the local region in the next frame and conducts correlations for each detection from the previous frame in short-term cues. Similarly, in [73], for each anchor object in the center frame, the cross-correlation is conducted between the anchor feature and local regions in adjacent frames. Built upon the single-shot detector (SSD) [168], [169] re-detects the targets via correlation with tracking anchors. SiamMOT [75] proposes the explicit motion model (EMM) to estimate the cross-channel correlation between the detected patch and local regions of the next frame, following the Faster R-CNN detector. [76] proposes a re-check network that conducts cross-correlation between previous embeddings and current feature maps. [77] proposes a global correlation network that estimates the correlation between features of historical tracks and the current feature map for box regression following the Faster R-CNN framework. [78] learns contrastive embeddings around temporal neighbors sampled from ROIs for both positive and negative instances, following the Faster R-CNN framework.

Attention via Transformers. With recent advances of visual Transformers, some approaches [42], [43], [159], [170] adopt Transformer in MOT, since transformers use pairwise attention that can fuse global information in the embedding and boost the tracking performance. The query-key mechanism plays a role of correlation in the tracking. The prediction can be obtained with the multi-head attention [171] that measures the correlation between feature maps and track queries. Specifically, TransTrack [170] uses track queries and object queries in the Transformer decoder for both track prediction and object detection. Similarly, [42], [43] also employ the Transformer decoder to estimate the correlation between previous tracks and current feature maps for prediction. PatchTrack [159] takes predicted tracks from Kalman Filtering as queries in the Transformer decoder to estimate the correlation between predicted tracks and feature maps from the encoder.

Reconstruction-based regression. Apart from approaches that use correlation for prediction, some works adopt reconstruction-based strategies with regression to learn the correlations for embedding. For example, built upon a variant of DLA-34 [134], SOTMOT [172] solves ridge regression coefficient from the previous frame for each object and uses the learned coefficient to predict the label for the next frame. As a result, discriminative features can be learned around neighbors in local regions.

Correlation-based embedding can learn the local-global correspondence. This type of embedding also can borrow the state-of-the-art methods from SOT problems. When targets are partially occluded and lost from detectors, such trackers are still able to continually track the objects. However, the drift becomes the major issue when long-time occlusion occurs, and trackers may gradually lose the targets.

3.6 Sequential Embedding

Another commonly used strategy to model temporal information in MOT is to use recurrent neural networks for sequential modeling. Such sequential embedding methods learn the dynamic update of transformations from the previous embedding to the current embedding. Embedding methods of this category can be formulated as follows,

$$\boldsymbol{z}_i^t = f(\boldsymbol{z}_i^{t-1}, \boldsymbol{x}_i^t), \tag{8}$$

where $\boldsymbol{z}_i^{t-1}$ represents the historical embeddings, $\boldsymbol{x}_i^t$ represents the input to the current state.

Pure motion. Some works use sequential embedding to model motion features. For example, [101] uses RNN for motion prediction. TrajE [173] adopts RNN to estimate a mixture of Gaussians that model the object trajectory. [27], [174] use LSTM to embed the motion information. [18] adopts LSTM for embedding and association.

Appearance incorporated. Some works also take consideration of appearance features in the sequential embedding. For example, [175] employs the convolutional gated recurrent unit (ConvGRU) to aggregate appearance features in the track memory. [176] proposes Recurrent Tracking Unit (RTU) to model long-term temporal information, in which RTU takes the old appearance feature template, the appearance feature of the current node, the old hidden state, and the state feature of the current node as input for embedding. [177] uses ConvGRU [178] to encode the entire frame for feature map estimation. [179] exploits bilinear LSTM [180] to encode the appearance of tracklet and trains a binary classifier on the positive and hard negative tracklets to learn discriminative embeddings. ArTIST [181] proposes the moving agent network, which is built upon a recurrent auto-encoder neural network that learns to reconstruct the tracklets of all moving agents with both appearance and motion cues.

Sequential embedding can learn the long dependency of the motion of targets. However, the standalone sequential embedding cannot achieve high performance. As a result, it is usually combined with other embedding methods, such as tracklet embedding, cross-track relational embedding, and single-frame detection embedding.

3.7 Tracklet Embedding

Following the tracking-by-detection paradigm, many approaches treat tracklets as individual units rather than detections, aiming at exploiting local and global temporal cues in the embedding. Typically, tracklet embedding can also be divided into two sub-categories: tracklet self-embedding and pairwise embedding.

3.7.1 Self-Embedding

Self embedding can be formulated as follows,

$$\boldsymbol{z}_i = f(\mathcal{T}_i), \tag{9}$$

where $\mathcal{T}_i = [\boldsymbol{D}_i^{t-\tau}; \boldsymbol{D}_i^{t-\tau+1}; ...; \boldsymbol{D}_i^{t}]$ is the i-th tracklet, containing sequential detections of the same object from local intermediate prediction.

Set-based. Some works treat each tracklet as a disordered set. Then learning the tracklet embedding is similar to the set classification problem. For example, [182] uses multi-head attention to aggregate features of individual detections and learns the tracklet embedding with the combination of triplet loss and cross-entropy loss for the association. [106] adopts contrastive loss between detection embedding and tracklet embedding using average features of detections. It also combines both labeled tracks and unlabeled tracks extracted from AVA-Kinetics [122], [123] in training to boost the tracking performance.

Temporal-based. To better model the temporal cues, many works also take the temporal order into the embedding. For example, [183] uses CNN and bi-directional convolutional LSTM to select representative detections for tracklet embedding. [184] adopts temporal convolution to model temporal attention [185] for feature aggregation to obtain tracklet embedding. [40] proposes a reconstruct-to-embed framework to embed the tracklet based on motion features with temporal convolution and GNN to learn the interaction among tracklets. TransMOT [41] uses a spatial-temporal graph Transformer encoder to encode tracklets and uses spatial graph decoder to estimate the association between tracklets and detections. [186] concatenates both averaged appearance features and the spatial-temporal features and then graph convolutional neural networks (GCN) [187] is further adopted for tracklet embedding.

Compared with other embedding methods, the tracklet self-embedding can incorporate motion information from nature. As a result, the appearance and motion features can be combined with a unified model. However, generating tracklets is tricky. The errors from the tracklet generation can cause degradation on the tracklet embedding.

3.7.2 Pairwise Embedding

Different from self-embedding, pairwise embedding jointly models the association relationship of the input tracklet pair. This kind of approach can be summarized as follows,

$$\boldsymbol{z}_{i,j} = f(\mathcal{T}_i, \mathcal{T}_j), \tag{10}$$

where $\boldsymbol{z}_{i,j}$ is the pairwise embedding of the tracklet $\mathcal{T}_i$ and $\mathcal{T}_j$.

Tracklet-tracklet pair. Specifically, [188] learns unary and pairwise potentials based on the CRF model [189], [190] using CNN and LSTM architecture, where each graph node represents a pair of tracklets. [191] uses a deep network for tracklet Re-ID that computes pairwise similarity scores between tracklets by jointly learning visual and spatial-temporal features. The network consists of a CNN that learns pairwise detection visual appearance and two bidirectional RNNs that learn spatial-temporal features. Then both two features are aggregated as the pairwise tracklet embedding. Following the multiple hypothesis tracking [192] framework, [193] measures the input tracklets whether they belong to the same track using CNN and LSTM to model appearance and motion separately in the proposed appearance evaluation network and motion evaluation network, respectively. TNT [194] takes account of temporal

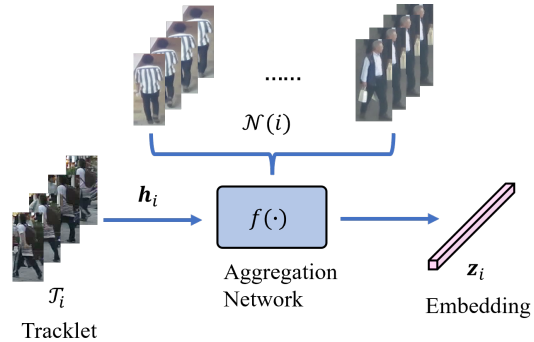

Fig. 7. Cross-track relational embedding.

orders of the input tracklet pair based on the temporal connectivity for pairwise embedding.

Tracklet-detection pair. Some works model the relation between a tracklet and a detection. In other words, $\mathcal{T}_j$ from Eq. (10) can only contain a single detection. For example, [195] learns a pair of detection with softmax as binary classification in the joint-inference network [126]. Then a detection is compared with all previous detections within the tracklet, and LSTM is adopted to measure the similarity in the association network. Inspired by the object relation module in [196], [197] proposes a spatial-temporal relation module to learn appearance, topology, and location cues, and the final similarity is measured for each tracklet-object pair. [180] learns the embedding of tracklet-detection pair with the proposed bilinear LSTM. [198] sequentially applies CNN and LSTM to obtain the tracklet embedding. Then a matching network is conducted to embed the concatenated features of the tracklet-detection pair for matching and association. [199] adopts a spatial attention network to obtain the feature of each pair of the detection and a frame of the tracklet. Bi-directional LSTM (Bi-LSTM) is exploited to learn the temporal feature of the SAN in the temporal attention network. Then the final embedding is learned for each tracklet-detection pair.

Tracklet pairwise embedding has similar properties as tracklet self-embedding methods. Besides that, it can also learn the discriminative relations and temporal order among different tracklets. Since each pair of tracklets needs to be embedded separately, the computational cost is huge, which is the major weakness.

3.8 Cross-Track Relational Embedding

Cross-track relational embedding aims at learning the object features based on the interactions with the neighboring tracks. The embedding can be represented as follows,

$$\boldsymbol{z}_i = f(\{\boldsymbol{h}_j | j \in i \cup \mathcal{N}(i)\}), \tag{11}$$

where $\boldsymbol{h}_j$ is the intermediate embedding of a detection or a tracklet; $\mathcal{N}(i)$ represents the connected neighbors of the i-th track defined in the approach.

Attention-based. The attention mechanism is one of the commonly used strategies to obtain track embeddings with the interactions of other tracks. For example, [175] proposes temporal-aware target attention and distractor attention for feature extraction. Transformer-based trackers also use attention to obtain cross-track embeddings. [42], [43] encode the track embeddings using the correlation with other tracks and objects via multi-head attention. [200] embeds both current-frame detections as well as the past detections via encoding the spatial-temporal context information based on Transformer with Attention Measurement Encoding. TransMOT [41] proposes a spatial-temporal graph transformer encoder to encode tracklets and uses a spatial graph decoder to estimate the association between tracklets and detections.

Graph-based. Graph-based methods are also widely employed for cross-track embedding. Specifically, [201] defines local graphs for each detected object and its k-nearest neighbors. The graph similarity model (GSM) is used for every two frames to measure the association via binary classification. [202] uses GCN to learn the interaction features of detections in two frames for the association. DeepMOT [137] proposes a deep Hungarian network that uses bi-directional RNN to model the association and defines differentiable dMOTA and dMOTP metrics for training. [203] models the motion graph network and appearance graph network separately. The graph is built for detections in adjacent two frames. The final similarity is measured with a weighted sum of motion and appearance. [174] uses GNN to model the association of detections in sequential two frames and relation update layer to measure the association as binary classification. Following the message passing network [204], [205], [206], [207], [6], [208] aggregate neighborhood nodes for detection embedding. [209] proposes an adapted graph neural network (AGNN) with GNN to associate tracklets and detection. [210] builds detection and tracklet graphs. Then the cross-graph GCN is adopted to associate detection and tracklets. Inspired by the object relation module [196], [197] designs a spatial-temporal relation module for learning appearance, topology, and location cues. The final similarity is then measured for the tracklet-object pair. [40] proposes a reconstruct-to-embed strategy to embed the tracklet based on motion features, and GCN is adopted to learn the interaction among tracklets. [186] concatenates both averaged appearance features and spatial-temporal features. Then deep aggregated features are learned via GCN [187]. ArTIST [181] proposes a moving agent network (MA-Net), where MA-Net is a recurrent auto-encoder neural network that learns to reconstruct the tracklets of all moving agents potentially interacting with the tracklet of interest. [5] exploits node feature aggregation based on GNN given two sequential frames.

Unlike other embedding strategies, cross-track embedding can learn the interaction among tracks. The discrimination between different IDs can be significantly exploited. It can also be combined with other embedding methods, like sequential and tracklet embedding, to enhance representation learning. However, a thorough analysis of the effectiveness of cross-tracking embedding has not been well studied, requiring more future research.

4 Benchmarks and Analysis

4.1 Datasets

We first review 11 widely used datasets for MOT, including KITTI [177], [211], [212], MOT15 [213], DukeMTMCT [117], MOT16-17 [143], PathTrack [124], UA-DETRAC [214], PoseTrack [215], [216], MOTS [37], CityFlow [20], KITTI MOTS [37], MOT20 [32], [144], nuScenes [217], Waymo [218], BDD100K [219], [220], and VisDrone [221], [222], [223], [224]. These datasets mainly focus on the person and vehicle tracking. Some are for general pedestrian tracking, and some are for traffic flow analysis and autonomous driving. The annotation includes 2D and 3D bounding boxes, pose and keypoint, and instance masks. The statistics of these datasets are summarized in Table 2. There are also other tracking datasets for specific tasks, such as HiEve [225], [226], DanceTrack [227], Omni-MOT [38], [228], Virtual KITTI [229], Apollo MOTS [111], TAO-person [230], WildTrack [231], and GMOT-40 [232]. Details of these datasets can be found in the reference.

TABLE 2
Statistics of MOT datasets. We use "-" to represent the missing information that is not released from the official site of the dataset.

Dataset	Year	#Track	#Box	Annotation
KITTI [211]	2012	-	-	2D box/3D box
MOT15 [213]	2015	1,221	101.3K	2D box/3D box
DukeMTMCT [117]	2016	6.7K	-	2D box
MOT16-17 [143]	2017	1,331	300.4K	2D box
PathTrack [124]	2017	16.3K	-	2D box
UA-DETRAC [214]	2017	8.2K	1.2M	2D box
PoseTrack [215]	2018	-	153.6K	Pose
VisDrone [223]	2018	-	1.8M	2D box
BDD100K [219]	2018	160K	4M	2D box/Mask
MOTS [37]	2019	228	26K	Mask
KITTI MOTS [37]	2019	-	-	Mask
CityFlow [20]	2019	666	230K	2D box
MOT20 [144]	2020	3,833	2.1M	2D box
Waymo [218]	2020	-	12.6M	2D box/3D box
nuScenes [217]	2020	-	1.4M	3D box

Several observations can be made in terms of the dataset collection over recent years: 1) More challenging, towards crowded scenarios with high object density. 2) More diverse annotation format, 2D and 3D boxes to masks and poses. 3) Large scales, either more annotations or more tracks and videos.

4.2 Evaluation Metrics

To evaluate the performance of MOT, CLEAR MOT [233], ID-based metrics [117], and HOTA [234] are widely used measurements.

CLEAR MOT [233] measures the multiple object tracking accuracy (MOTA) and multiple object tracking precision (MOPT) between the detected boxes and ground truth boxes. The proposed MOTA and MOTP are based on the matched pairs, misses, false positives, and mismatches. However, MOTA overemphasizes detection performance rather than association.

ID-based metrics [117] compute the truth-to-result match that measures the tracking performance by how long the tracker correctly identifies targets. Specifically, a bipartite match associates one ground-truth trajectory to exactly one computed trajectory by minimizing the number of mismatched frames over all the available data-true and computed. Standard measures such as precision, recall, and F1-score are built on top of this truth-to-result match, namely

IDP, IDR, and IDF1. However, IDF1 overemphasizes association performance rather than detection.

A higher order metric, namely higher order tracking accuracy (HOTA), is proposed in [234] for evaluating MOT performance. HOTA explicitly balances the effect of performing accurate detection, association, and localization into a single unified metric for comparing trackers. HOTA decomposes into a family of sub-metrics that are able to evaluate each of five basic error types (detection recall, detection precision, association recall, association precision, and localization accuracy) separately, which enables clear analysis of tracking performance.

There are also other metrics for evaluation. For example, PR-MOTA is proposed in the UA-Detrac dataset [214] and takes account of the detection score along PR-curve with the MOTA metric for MOT measurement. In the nuScenes dataset [217], sAMOTA is adopted as the primary evaluation metric for the 3D MOT benchmark, in which an updated formulation of MOTA is used to adjust for the respective recall. [37] introduces the soft multi-object tracking and segmentation accuracy (sMOTSA) for MOTS-based tracking measurement. [235] proposes the SAIDF evaluation measure that focuses more on identity issues and fixes the insensibility and high computational cost problems of the previous measures, such as MOTA and IDF1. Furthermore, [236] introduces local metrics, namely LIDF1 and ALTA, that are parametrized by a temporal horizon and thereby reveal the temporal ranges at which association errors occur. Such metrics provide more insights from different aspects for MOT evaluation.

4.3 In-Depth Analysis on State-of-the-Art Methods

We review the state-of-the-art embedding methods of MOT on widely used datasets, including MOT17, MOT20, and KITTI. We include methods published in top CV venues over the past three years. We mainly focus on three evaluation metrics: MOTA, IDF1, and HOTA. The performances on three benchmarks are reported in Table 3, Table 4, and Table 5. The embedding methods in the table include "Patch-Level Box Image Embedding" (Patch), "Single-Frame Detection Embedding" (S-Fr), "Single-Frame Detection Embedding" (X-Fr), "Correlation-Based Embedding" (Corre), "Sequential Embedding" (Seq), "Sequential Embedding" (Tracklet) and "Sequential Embedding" (X-Track). The used detection can be Public (Pub.) or Private (Private). Modality (Mod.) in the KITTI benchmark includes Vision (V) and LiDAR (L). The Object (Obj.) category includes Car (C) and Person (P). We analyze the top performance for each category of embedding methods as follows.

TABLE 3
Summary of embedding methods on MOT17 benchmark.

Emb. Method	Ref.	Year	Venue	Det.	MOTA	IDF1	HOTA
Patch	[102]	2017	CVPRW	Pub.	50.0	51.3	41.3
	[105]	2018	CoRR	Pub.	51.5	46.9	38.5
	[103]	2018	ICME	Pub.	50.9	52.7	41.2
	[100]	2018	AVSS	Pub.	48.3	51.1	40.3
	[72]	2019	ArXiv	Pub.	54.7	62.3	47.1
	[125]	2019	Access	Pub.	48.6	47.9	38.4
	[68]	2020	ArXiv	Pub.	61.7	58.1	46.9
	[98]	2020	ICRAI	Pub.	49.7	51.5	39.0
	[70]	2021	J-VCIR	Pub.	46.8	54.1	41.5
	[71]	2021	IVC	Priv.	77.0	72.0	59.7
	[106]	2021	ArXiv	Priv.	73.3	73.2	59.8
S-Fr	[15]	2019	ICCV	Pub.	56.3	55.1	44.8
	[33]	2021	AI	Pub.	60.1	58.8	47.2
	[35]	2021	ArXiv	Priv.	73.8	74.7	61.0
	[34]	2021	IJCV	Priv.	73.7	72.3	59.3
	[132]	2022	Neuroc.	Priv.	73.5	70.2	58.7
X-Fr	[12]	2020	ECCV	Pub.	61.5	59.6	48.2
	[146]	2019	T-PAMI	Priv.	52.4	49.5	39.3
	[14]	2020	CVPR	Priv.	63.0	58.6	48.0
	[13]	2020	ECCV	Priv.	66.6	57.4	49.0
	[12]	2020	ECCV	Priv.	67.8	64.7	52.2
	[5]	2021	ICRA	Priv.	73.2	66.5	55.2
	[147]	2021	ArXiv	Priv.	73.2	62.2	54.5
	[159]	2022	ArXiv	Priv.	73.6	65.2	53.9
Corre	[73]	2019	ICCV	Pub.	52.0	48.7	-
	[163]	2020	AAAI	Pub.	49.5	51.8	41.5
	[170]	2020	ArXiv	Priv.	75.2	63.5	54.1
	[165]	2021	CVPR	Priv.	76.5	73.6	60.7
	[78]	2021	CVPR	Priv.	68.7	66.3	53.9
Seq	[101]	2019	Neuroc.	Pub.	45.1	43.2	-
	[177]	2021	ICCV	Pub.	73.1	67.2	54.2
	[173]	2021	ArXiv	Pub.	67.8	61.4	50.0
	[179]	2021	CVPR	Pub.	51.5	54.9	41.3
	[176]	2021	ICCV	Priv.	74.9	75.0	62.0
	[177]	2021	ICCV	Priv.	73.8	68.9	55.5
Tracklet	[191]	2018	ArXiv	Pub.	54.1	48.4	46.8
	[199]	2018	ECCV	Pub.	48.2	55.7	42.5
	[180]	2018	ECCV	Pub.	47.5	51.9	41.0
	[194]	2019	MM	Pub.	51.9	58.1	44.9
	[197]	2019	ICCV	Pub.	50.9	56.0	42.6
	[193]	2020	T-IP	Pub.	54.9	63.1	48.4
	[183]	2020	PR	Pub.	54.2	52.6	41.5
	[188]	2020	T-CSVT	Pub.	53.1	53.7	42.2
	[186]	2021	CVPR	Pub.	59.0	66.8	51.5
	[198]	2021	NeurIPS	Pub.	56.8	58.3	46.4
	[195]	2021	IS	Pub.	50.3	53.5	42.0
	[109]	2021	J-VCIR	Pub.	45.4	39.9	34.0
	[182]	2022	T-MM	Pub.	61.5	63.3	50.5
X-Track	[6]	2020	CVPR	Pub.	58.8	61.7	49.0
	[202]	2020	ArXiv	Pub.	57.3	56.3	45.4
	[201]	2020	IJCAI	Pub.	56.4	57.8	45.7
	[137]	2020	CVPR	Pub.	53.7	53.8	42.4
	[209]	2020	ArXiv	Priv.	76.2	68.0	57.9
	[5]	2021	ICRA	Priv.	73.2	66.5	55.2

Patch-Level Image Embedding. [71], [106] achieve the top performance on MOT17 and MOT20 datasets with private detections. They outperform other methods significantly on MOTA, IDF1, and HOTA. Specifically, [71] proposes a post-processing strategy that refines the mis-associated tracklets with the retraining of an appearance encoder, leading to the top performance on patch-level image embedding. However, since [71] is a post-processing method, it is also highly dependent on the baseline tracking methods. [106] benefits from mining unlabeled tracks with pseudo labels to boost feature learning, resulting in a good performance. On KITTI benchmark, [237], [238] achieve leading performance, where [238] learns local descriptors and [237] follows a simple fusion of appearance and LiDAR features. For patch-level embedding, they usually follow the conventional tracking-by-detection scheme, in which detectors and trackers can be trained separately. With retrained detectors using large-scale training data, simple embedding methods can be effective with reliable detections.

Single-Frame Joint Detection Embedding. [34], [35],

TABLE 4
Summary of embedding methods on MOT20 benchmark.

Emb. Method	Ref.	Year	Venue	Det.	MOTA	IDF1	HOTA
Patch	[68]	2020	ArXiv	Pub.	53.6	50.6	41.7
	[70]	2021	J-VCIR	Pub.	44.7	43.5	35.6
	[71]	2021	IVC	Priv.	77.4	73.1	61.2
	[106]	2021	ArXiv	Priv.	65.2	70.1	55.3
S-Fr	[15]	2019	ICCV	Pub.	52.6	52.7	42.1
	[33]	2021	AI	Pub.	59.3	59.1	47.1
	[35]	2021	ArXiv	Priv.	67.2	70.5	56.5
	[34]	2021	IJCV	Priv.	61.8	67.3	54.6
	[132]	2022	Neuroc.	Priv.	68.6	69.4	56.2
X-Fr	[147]	2021	ArXiv	Pub.	61.0	49.8	43.5
	[5]	2021	ICRA	Priv.	67.1	67.5	53.6
Corre	[170]	2020	ArXiv	Priv.	65.0	59.4	48.9
Tracklet	[186]	2021	CVPR	Pub.	56.3	62.5	49.0
	[182]	2022	T-MM	Pub.	54.6	53.4	42.5
X-Track	[6]	2020	CVPR	Pub.	57.6	59.1	46.8
	[202]	2020	ArXiv	Pub.	54.5	49.0	40.2

TABLE 5
Summary of embedding methods on KITTI MOT benchmark.

Emb. Method	Ref.	Year	Venue	Mod.	Obj.	MOTA	HOTA
Patch	[128]	2019	ArXiv	V	C	75.8	60.9
	[237]	2020	IROS	V&L	C	85.1	69.6
	[238]	2020	ACCV	V	C	87.8	68.5
	[238]	2020	ACCV	V	P	68.0	50.9
	[237]	2020	IROS	V&L	P	45.3	34.2
S-Fr	[28]	2019	ICCV	V	C	84.3	73.2
X-Fr	[239]	2019	ICCV	V&L	C	83.2	62.1
	[240]	2020	RAL	L	C	67.6	57.2
	[12]	2020	ECCV	V	C	88.8	73.0
	[241]	2021	IJCAI	L	C	91.7	80.9
	[27]	2021	CVPR	V	C	88.4	74.2
	[242]	2021	RAL	L	C	84.5	72.2
	[243]	2021	IROS	V&L	C	85.4	70.7
	[12]	2020	ECCV	V	P	53.8	40.4
Corre	[73]	2019	ICCV	V	C	75.9	52.6
	[244]	2021	ArXiv	V	C	85.9	72.8
	[78]	2021	CVPR	V	C	84.9	68.5
	[78]	2021	CVPR	V	P	55.6	41.1
	[244]	2021	ArXiv	V	P	51.8	41.1
Seq	[28]	2019	ICCV	V	C	84.3	73.2
	[177]	2021	ICCV	V	C	91.3	78.0
	[177]	2021	ICCV	V	P	66.0	48.6
Tracklet	[40]	2021	ICCV	V	C	87.6	73.1
X-Track	[40]	2021	ICCV	V	C	87.6	73.1
	[208]	2021	ArXiv	V	C	87.3	72.3
	[6]	2020	CVPR	V	P	46.2	45.3
	[208]	2021	ArXiv	V	P	52.1	39.4

[132] achieve the leading performance on MOT17 and MOT20 benchmarks with private detections. These three methods follow the decoupling strategy for learning the Re-ID features to improve performance. Such single-frame joint detection embedding methods do not rely on temporal relations, so that they can efficiently learn robust features even with single-image based detection datasets. That is the reason why this kind of embedding strategy can attract more and more researchers recently.

Cross-Frame Joint Embedding. [5] outperforms other embedding methods in this category on both IDF1 and HOTA on MOT17 and MOT20 benchmarks. Specifically, [5] aggregates the cross-frame embedding in the head subnetwork and employs cross-track embedding to learn discriminative relationships among different objects. With the combination of cross-frame and cross-track embedding strategy, [5] achieves the leading performance. On the KITTI benchmark, [241] proposes a tracklet proposal generation network with refinement using cross-frame joint embedding strategy on the 3D point cloud as input, achieving the best performance on both MOTA and HOTA. Cross-frame embedding can benefit from spatial-temporal feature aggregation. Besides, it can combine with other embedding techniques, such as sequential embedding and cross-track embedding, to further improve the performance. However, since multiple frames are treated as one input sample for training in this category, it works as the most end-to-end embedding method, and it usually requires more training data than other types of embedding methods.

Correlation-based embedding. The top performance of this category is from [165] on MOT17, [170] on MOT20, and [244] on KITTI. Specifically, [165] learns spatial and temporal correlations with neighborhood regions for embedding. [170] measures the correlation with the attention mechanism through Transformer architecture. [244] builds contrastive pairs to learn object identities with sampled neighbors. Such correlation-based embedding can learn discriminative ID features either locally or globally to boost the tracking performance. They can also borrow the learning strategies from the SOT task.

Sequential Embedding. [176], [177] achieve the best performance on MOT17 and KITTI, respectively. [176] follows the tracking-by-detection scheme and proposes a general multiple nodes tracking framework with a recurrent tracking unit that can be extended to most trackers and adapts to various scenarios to improve the performance of the baseline tracker. [177] uses a mix of synthetic and real data with ConvGRU that can outperform the state-of-the-art methods. Standalone sequential embedding usually cannot achieve leading performance. It requires either good feature extractors or reliable detectors as support. It can also combine with other embedding methods to improve the performance.

Tracklet Embedding. [40], [182], [186] achieve top performance on MOT17, MOT20 and KITTI, respectively. [182] designs a post-processing strategy with tracklet embedding that can be extended to baseline trackers. [186] learns spatial-temporal features for pairwise tracklet embedding. [40] proposes a reconstruct-to-embed strategy and employs cross-track relations for tracklet embedding. Since [182], [186] do not refine the public detections, the performance is not competitive with private trackers. To further boost the performance, reliable detections are needed for tracklet embedding methods.

Cross-Track Relational Embedding. [6], [40], [209] are the top trackers on MOT17, MOT20, and KITTI, respectively. In this embedding category, discriminative embeddings can be learned by graph neural networks to measure the relationships among objects. This embedding strategy can be beneficial for scenarios with multiple similar objects. It can be combined with other embedding methods as well, such as patch-level embedding, tracklet embedding, and cross-frame joint embedding.

In summary, different embedding strategies have their own benefits. Combining multiple embedding methods can usually boost the performance. Besides, learning with extra training data, designing appropriate embedding architecture, and tracking with different association approaches also affect the final performance. An appropriate embedding method that fits the entire tracking framework would be worth thinking for researchers to propose novel trackers.

5 Future Directions of MOT Embedding Methods

5.1 Under-Investigated Areas

We discuss the trends and potential future directions for MOT embedding methods that are not well investigated from five different aspects, including non-fully supervised learning, generalization and domain adaptation, embedding for crowded scenes, multi-view collaboration, and multi-modal MOT.

5.1.1 Non-Fully Supervised Learning

Most existing embedding methods in MOT use a fully supervised training framework with object boxes/masks and track IDs. However, it is expensive to annotate a large amount of video data from diverse scenarios. Utilizing non-fully annotated video data is in high demand in the MOT area.

Some works try to adopt weakly supervised and semi-supervised learning frameworks to alleviate the annotation cost. Typically, weakly supervised learning use the weak label in the annotation for each sample, while semi-supervised learning combines labels and unlabels samples in training. For example, [245] employs weakly instance segmentation by Grad-CAM heatmaps [246] to extract partial foreground masks for the MOTS task. In place of tracking annotations, [247] first hallucinates videos from images with bounding box annotations using zoom-in/out motion transformations to obtain free tracking labels. [248] generates pseudo labels for successive frames with the association for cell tracking. [106] adopts contrastive representation learning between detection and sub-tracklet embeddings, combining both labeled tracks and unlabeled tracks in training. Some works [68] claim that they propose unsupervised trackers that do not use ground truth tracking labels. However, they assume the box annotation or detection results are available. As a result, these are actually weakly supervised approaches. For example, [68] generates tracklets by SORT tracker [121] and trains the Re-ID embeddings with the pseudo labels. [132] designs an unsupervised Re-ID framework based on strong and weak supervision of the association cues from two frames following FairMOT [34]. [249] proposes a transductive interactive self-training method to adapt the tracking model to unseen crowded scenes with unlabeled testing data by means of teacher-student interactive learning. However, these weakly supervised and semi-supervised approaches usually generate pseudo tracking labels with augmentation and simple trackers that could bring much tracking error. Recently, [250] proposes a novel masked warping loss that drives the network to indirectly learn how to track objects through a video, trained only with the bounding box information. This type of loss only considers a very short period of temporal consistency. As a result, it cannot achieve competitive performance compared with supervised approaches.

Some works [198], [251] follow self- and unsupervised learning framework in MOT. This kind of work takes consideration of prior consistency and constraints in embedding learning. Specifically, [198] proposes a self-supervised tracker by using cross-input consistency, in which two distinct inputs are constructed for the same sequence of video by hiding different information about the sequence in each input. [251] proposes a tracking-by-animation (TBA) framework, where a differentiable neural model tracks objects from input frames and then animates these objects into reconstructed frames, which achieves both label-free and end-to-end learning of MOT. [252] follows the TBA framework for noisy environments.

5.1.2 Training with Synthetic Data

Since fully annotated real video data is hard to collect, many works [229], [253], [254] try to utilize synthetic data for training trackers. The annotations such as box, mask and track ID, can be naturally obtained from synthetic data. For example, [229] shows pre-training on virtual data can improve performance. [253] introduces a large-scale synthetic data engine named MOTX to generate synthetic data for MOT training and achieve competitive results with the trackers trained using real data. In addition, [254] shows that better performance can be achieved with synthetic data on some widely known trackers such as Tracktor [15], Track R-CNN [37], Lift_T [255], MPNTrack [6], and CenterTrack [12]. These works prove the efficiency of utilizing synthetic data for the MOT task. However, how to reduce the domain gap between synthetic and real data and design a learning framework with the generalization for unseen diverse scenarios is still under-explored.

5.1.3 Multi-View Collaborated MOT

In recent years, there have been a few works about multi-view MOT [256], [257], [258], [259], [260], [261], [262], [263], [264], where multiple objects are tracked from several different overlapping views simultaneously, aiming to address occlusion issues with multi-view geometry and consistency. The embedding framework is learned for both cross-frames and cross-views. The multi-view also provides the depth information, enabling tracking in the 3D space. Specifically, [260] proposes a hierarchical composition model and re-formulates multi-view MOT as a problem of compositional structure optimization. [263] models the data similarity using appearance, motion, and spatial reasoning and formulates the multi-view MOT as a joint optimization problem solved by constrained integer programming. [261] formulates multi-view MOT as a constrained mixed-integer programming problem and effectively measures subjects' similarity over time and across views. [257] proposes a spatial-temporal association network with two designed self-supervised learning losses, including a symmetric-similarity loss and a transitive-similarity loss, to associate the multiple humans over time and across views. In [259], tracklets are matched to multi-camera trajectories by solving a global lifted multicut formulation that incorporates short and long-range temporal interactions on tracklets located in the same camera as well as inter-camera ones. In addition, [262] presents a novel solution for multi-human 3D pose estimation and tracking simultaneously from multiple calibrated camera views.

However, there are still unresolved issues for such settings. First, with limited multi-view collaborated dataset, cross-view Re-ID embedding is not robust and causes the degradation for tracking performance. Second, it is difficult to define an end-to-end joint detection and tracking framework that can take multiple views with variant spatial-temporal relations. Third, how to better utilize the cross-frame and cross-view geometry consistency is still not well-explored.

5.1.4 Multi-Modal MOT

Except for vision-based trackers, other modalities can also be adopted in MOT. Since LiDAR has a good measure for depth, LiDAR-based trackers [241], [265], [266], [267], [268] have become popular in autonomous driving scenarios. As different sensors (*e.g.*, RGB camera, LiDAR, radar) get increasingly used together, the multi-modal MOT [239], [243], [269], [270], [271], [272], [273], [274] starts to attract research attention. One benefit is that multiple sensors increase the diversity of object representations, which provides higher association reliability across objects from different timestamps. Different modalities have their own advantages and disadvantages. In a tracking-by-detection framework, fusing data from multiple modalities would ideally improve tracking performance than using a single modality.

For example, [269] integrates a real-time multi-modal laser/RGB-D people tracking framework for moving platforms. [239] proposes a generic sensor-agnostic multi-modality MOT framework, where each modality is capable of performing its role independently to preserve reliability, and could further improve its accuracy through a novel multi-modality fusion module. In [270], features of RGB images and 3D LiDAR points are extracted separately before they fully interact with each other through a cross-modal attention mechanism for inter-frame proposals association. [271] learns how to fuse features from 2D images and 3D LiDAR point clouds to capture the appearance and geometric information of an object. [243] presents an efficient multi-modal MOT framework with online joint detection and tracking schemes and robust data association for autonomous driving applications. propose a fully [273] proposes an end-to-end network for joint object detection and tracking with a center-based radar-camera fusion algorithm. [274] presents a novel self-supervised MM-DistillNet framework consisting of multiple teachers that leverage diverse modalities, including RGB, depth, and thermal images, to simultaneously exploit complementary cues and distill knowledge into a single audio student network.

However, there are still unresolved issues for multi-modal MOT. First, the embedding strategy with multi-modality is not well explored, following the simple fusion scheme. Second, except for LiDAR and radar, other modalities are not frequently used. For example, how to combine sound, text, infrared sensors, and other modalities should be explored for some special cases [275], [276]. Third, addressing missing or noisy modalities in MOT is also one of the major challenges.

5.1.5 MOT with Reinforcement Learning

In recent years, deep reinforcement learning has gained significant successes in various vision applications, such as object detection [277], face recognition [278], image super-resolution [279] and object search [280]. However, deep reinforcement learning remains under-explored in the MOT area, with a limited number of works found [281], [282], [283], [284]. For example, [281] presents a novel, multi-agent reinforcement learning formulation of MOT that treats creating, propagating, and terminating object tracks as actions in a sequential decision-making problem. [283] develops a deep prediction-decision network, which simultaneously detects and predicts objects under a unified network via deep reinforcement learning.

There are several challenges in deep reinforcement learning based MOT frameworks. First, how to define the space of state and action is crucial. The defined states and actions should be implicit and not annotated with ground truth. Otherwise, it may turn out to become simple supervised learning approaches. Besides that, the defined space should assist the tracking performance. Rather than simple states such as “tracked” and “lost”, other more complex behaviors can be models, such as “leaving”, “start walking”, etc. Second, the definition of reward functions is also important. Reward functions should reflect the relationship between chosen actions and tracking performance based on the defined space. Deep reinforcement learning may gain more potential in the MOT field with carefully designed space and reward functions.

5.2 Towards Future Embedding Learning in MOT

5.2.1 Meta-Learning

The field of meta-learning, or learning-to-learn, has seen a dramatic rise in interest in recent years [285], [286]. Contrary to conventional approaches in machine learning, where tasks are solved from scratch using a fixed learning algorithm, meta-learning aims to improve the learning algorithm itself, given the experience of multiple learning episodes. This “learning-to-learn” [287] can lead to a variety of benefits such as improved data and compute efficiency, and it is better aligned with human and animal learning [288], where learning strategies improve both on a lifetime and evolutionary timescales [288], [289], [290]. Meta-learning has been exploited in many computer vision tasks, such as few-shot object detection [291], [292] and segmentation [293], [294], [295]. However, meta-learning is still under-explored in the MOT task.

Meta-learning can be a potential novel direction in MOT with the learning-to-learn framework. First, meta-learner can be adopted to optimize the hyper-parameters, such as the detection and tracking threshold, multi-task loss weighting, and pseudo label generation. Second, the meta-learning can be used for generalization and adaptation purposes to assist MOT for diverse scenarios.

5.2.2 Learning with Auxiliary Tasks

Auxiliary learning [296] is a method to improve the ability of a primary task by training on additional auxiliary tasks alongside the primary task. The sharing of embeddings across tasks results in additional relevant features being available, which otherwise would not have been learned from training only on the primary task. The broader support of these features, across new interpretations of input data, then allows for better generalization of the primary task. Auxiliary learning is similar to multi-task learning [297], except that only the performance of the primary task is of importance, and the auxiliary tasks are included purely to assist the primary task. Applying related learning tasks is one straightforward approach to assist primary tasks. [298] applies auxiliary supervision with phoneme recognition at intermediate low-level representations to improve the performance of conversational speech recognition. [299] chooses auxiliary tasks which can be obtained with low efforts, such as global descriptions of a scene, to boost the performance for single scene depth estimation and semantic segmentation. By carefully choosing a pair of learning tasks, we may also perform auxiliary learning without ground truth labels in an unsupervised manner. [300] introduces a method for improving agent learning in Atari games by building unsupervised auxiliary tasks to predict the onset of immediate rewards from a short historical context. [301], [302] proposes image synthesis networks to perform unsupervised monocular depth estimation by predicting the relative pose of multiple cameras. [303] proposes to use cosine similarity as an adaptive task weighting to determine when a defined auxiliary task is useful. [157] explores auxiliary training in the context of keypoint embedding representation learning.

To learn embeddings for the MOT task, the defined auxiliary tasks should help improve the discrimination among different objects for the primary task. For example, incorporating unlabeled videos with an unsupervised loss for auxiliary learning; learning pose based features to improve the discriminative embeddings of objects; transferring image-based Re-ID features in the embedding models. It would largely broaden the view of researchers with such explorations.

5.2.3 Large-Scale Pre-Training

Self-supervised learning has gained popularity because of its ability to avoid the cost of annotating large-scale datasets [304]. It can adopt self-defined pseudo labels as supervision and use the learned representations for several downstream tasks. With recent self-supervised learning approaches, such as SimCLR [305], SwAV [306] and MoCo [307], and video-based tasks [308], [309], [310], [311], large-scale pre-training show great power in computer vision tasks. The pre-training aims at learning the representations with self-supervised pretext tasks with pseudo labels. The learned model from the pretext task can be used for downstream tasks such as classification, detection, and segmentation.

Since unlabeled video data can be easily obtained, large-scale pre-training can be a good opportunity for learning embeddings for MOT. However, two main issues should be addressed in such a direction. First, defining appropriate pretext tasks to learn object-level embeddings with spatial-temporal information is critical for MOT. Unlike video-level classification tasks, such as anomaly detection [312], [313] and action recognition [62], [314], object-level embeddings play an essential role in MOT. Second, how to transfer the pre-training model to the downstream tasks with track IDs is unclear. If these two issues can be addressed, the pre-training approaches will be beneficial to the MOT task with huge unlabeled video data.

5.2.4 Other Future Directions

There are other future directions for embedding learning methods in MOT, summarized below.

- Distilling knowledge for embedding learning from other tracking related models, such as image-based Re-ID models and detection model.
- Learning cross-domain embeddings to bridge the training and testing distribution differences.
- Mining priors, constraints and consistencies, such as enter-leave consistency (counting consistency), geometry consistency, and ego-motion consistency.
- Estimating implicit object behavior status to boost embedding.
- Reasoning and causality learning for object trajectory estimation.

6 Conclusion

This paper presents a comprehensive survey with an in-depth analysis for embedding methods in multi-object tracking (MOT). We first review the widely used embedding methods in MOT from seven aspects and provide a detailed summary for each embedding category. Then widely used MOT datasets and evaluation metrics are summarized. Besides that, an in-depth analysis of state-of-the-art methods is provided. We also point our several under-investigated areas and future worth-exploring research directions in the last section, hoping to inspire more thinking of the embedding strategies in MOT.

References

[1] Z. Tang, G. Wang, H. Xiao, A. Zheng, and J.-N. Hwang, "Single-camera and inter-camera vehicle tracking and 3d speed estimation based on fusion of visual and semantic features," in *CVPR workshops*, 2018, pp. 108–115.

[2] S. Tang, M. Andriluka, B. Andres, and B. Schiele, "Multiple people tracking by lifted multicut and person re-identification," in *CVPR*, 2017, pp. 3539–3548.

[3] A. Milan, K. Schindler, and S. Roth, "Multi-target tracking by discrete-continuous energy minimization," *IEEE TPAMI*, vol. 38, no. 10, pp. 2054–2068, 2015.

[4] R. Kumar, G. Charpiat, and M. Thonnat, "Multiple object tracking by efficient graph partitioning," in *ACCV*, 2014, pp. 445–460.

[5] Y. Wang, K. Kitani, and X. Weng, "Joint object detection and multi-object tracking with graph neural networks," in *ICRA*, 2021, pp. 13 708–13 715.

[6] G. Brasó and L. Leal-Taixé, "Learning a neural solver for multiple object tracking," in *CVPR*, 2020, pp. 6247–6257.

[7] C. Shan, C. Wei, B. Deng, J. Huang, X.-S. Hua, X. Cheng, and K. Liang, "Fgagt: Flow-guided adaptive graph tracking," *arXiv e-prints*, pp. arXiv–2010, 2020.

[8] X. Weng, Y. Wang, Y. Man, and K. M. Kitani, "Gnn3dmot: Graph neural network for 3d multi-object tracking with 2d-3d multi-feature learning," in *CVPR*, 2020, pp. 6499–6508.

[9] Q. Zhou, B. Zhong, Y. Zhang, J. Li, and Y. Fu, "Deep alignment network based multi-person tracking with occlusion and motion reasoning," *IEEE TMM*, vol. 21, no. 5, pp. 1183–1194, 2018.

[10] Q. Bao, W. Liu, Y. Cheng, B. Zhou, and T. Mei, "Pose-guided tracking-by-detection: Robust multi-person pose tracking," *IEEE TMM*, vol. 23, pp. 161–175, 2020.

[11] J. Shen, D. Yu, L. Deng, and X. Dong, "Fast online tracking with detection refinement," *IEEE TITS*, vol. 19, no. 1, pp. 162–173, 2017.

[12] X. Zhou, V. Koltun, and P. Krähenbühl, "Tracking objects as points," in *ECCV*, 2020, pp. 474–490.

[13] J. Peng, C. Wang, F. Wan, Y. Wu, Y. Wang, Y. Tai, C. Wang, J. Li, F. Huang, and Y. Fu, "Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking," in *ECCV*, 2020, pp. 145–161.

[14] B. Pang, Y. Li, Y. Zhang, M. Li, and C. Lu, "Tubetk: Adopting tubes to track multi-object in a one-step training model," in *CVPR*, 2020, pp. 6308–6318.

[15] P. Bergmann, T. Meinhardt, and L. Leal-Taixe, "Tracking without bells and whistles," in *ICCV*, 2019, pp. 941–951.

[16] Z. Wang, L. Zheng, Y. Liu, Y. Li, and S. Wang, "Towards real-time multi-object tracking," in *ECCV*, 2020, pp. 107–122.

[17] D. Y. Kim and M. Jeon, "Data fusion of radar and image measurements for multi-object tracking via kalman filtering," *Information Sciences*, vol. 278, pp. 641–652, 2014.

[18] A. Milan, S. H. Rezatofighi, A. Dick, I. Reid, and K. Schindler, "Online multi-target tracking using recurrent neural networks," in *AAAI*, 2017.

[19] Y. Lu, C. Lu, and C.-K. Tang, "Online video object detection using association lstm," in *ICCV*, 2017, pp. 2344–2352.

[20] Z. Tang, M. Naphade, M.-Y. Liu, X. Yang, S. Birchfield, S. Wang, R. Kumar, D. Anastasiu, and J.-N. Hwang, "Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification," in *CVPR*, 2019, pp. 8797–8806.

[21] G. Wang, X. Yuan, A. Zheng, H.-M. Hsu, and J.-N. Hwang, "Anomaly candidate identification and starting time estimation of vehicles from traffic videos." in *CVPR Workshops*, 2019, pp. 382–390.

[22] H.-M. Hsu, Y. Wang, and J.-N. Hwang, "Traffic-aware multi-camera tracking of vehicles based on reid and camera link model," in *ACM MM*, 2020, pp. 964–972.

[23] Y. Wu, D. Kong, S. Wang, J. Li, and B. Yin, "An unsupervised real-time framework of human pose tracking from range image sequences," *IEEE TMM*, vol. 22, no. 8, pp. 2177–2190, 2019.

[24] R. Gu, G. Wang, and J.-N. Hwang, "Efficient multi-person hierarchical 3d pose estimation for autonomous driving," in *MIPR*, 2019, pp. 163–168.

[25] R. Gu, G. Wang, Z. Jiang, and J.-N. Hwang, "Multi-person hierarchical 3d pose estimation in natural videos," *IEEE TCSVT*, vol. 30, no. 11, pp. 4245–4257, 2019.

[26] A. Jalal, M. Mahmood, and A. S. Hasan, "Multi-features descriptors for human activity tracking and recognition in indoor-outdoor environments," in *IBCAST*, 2019, pp. 371–376.

[27] M. Chaabane, P. Zhang, J. R. Beveridge, and S. O'Hara, "Deft: Detection embeddings for tracking," *arXiv preprint arXiv:2102.02267*, 2021.

[28] H.-N. Hu, Q.-Z. Cai, D. Wang, J. Lin, M. Sun, P. Krahenbuhl, T. Darrell, and F. Yu, "Joint monocular 3d vehicle detection and tracking," in *ICCV*, 2019, pp. 5390–5399.

[29] G. Wang, J.-N. Hwang, K. Williams, and G. Cutter, "Closed-loop tracking-by-detection for rov-based multiple fish tracking," in *CVAUI*, 2016, pp. 7–12.

[30] M.-C. Chuang, J.-N. Hwang, J.-H. Ye, S.-C. Huang, and K. Williams, "Underwater fish tracking for moving cameras based on deformable multiple kernels," *IEEE TSMC*, vol. 47, no. 9, pp. 2467–2477, 2016.

[31] M. Dawkins, L. Sherrill, K. Fieldhouse, A. Hoogs, B. Richards, D. Zhang, L. Prasad, K. Williams, N. Lauffenburger, and G. Wang, "An open-source platform for underwater image and video analytics," in *WACV*, 2017, pp. 898–906.

[32] P. Dendorfer, A. Osep, A. Milan, K. Schindler, D. Cremers, I. Reid, S. Roth, and L. Leal-Taixé, "Motchallenge: A benchmark for single-camera multiple target tracking," *IJCV*, vol. 129, no. 4, pp. 845–881, 2021.

[33] J. Yang, H. Ge, J. Yang, Y. Tong, and S. Su, "Online multi-object tracking using multi-function integration and tracking simulation training," *AI*, pp. 1–21, 2021.

[34] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "Fairmot: On the fairness of detection and re-identification in multiple object tracking," *IJCV*, vol. 129, no. 11, pp. 3069–3087, 2021.

[35] E. Yu, Z. Li, S. Han, and H. Wang, "Relationtrack: Relation-aware multiple object tracking with decoupled representation," *arXiv preprint arXiv:2105.04322*, 2021.

[36] Z. Lu, V. Rathod, R. Votel, and J. Huang, "Retinatrack: Online single stage joint detection and tracking," in *CVPR*, 2020, pp. 14 668–14 678.

[37] P. Voigtlaender, M. Krause, A. Osep, J. Luiten, B. B. G. Sekar, A. Geiger, and B. Leibe, "Mots: Multi-object tracking and segmentation," in *CVPR*, 2019, pp. 7942–7951.

[38] S. Sun, N. Akhtar, X. Song, H. Song, A. Mian, and M. Shah, "Simultaneous detection and tracking with motion modelling for multiple object tracking," in *ECCV*, 2020, pp. 626–643.

[39] X. Wan, S. Zhou, J. Wang, and R. Meng, "Multiple object tracking by trajectory map regression with temporal priors embedding," in *ACM MM*, 2021, pp. 1377–1386.

[40] G. Wang, R. Gu, Z. Liu, W. Hu, M. Song, and J.-N. Hwang, "Track without appearance: Learn box and tracklet embedding with local and global motion patterns for vehicle tracking," in *ICCV*, 2021, pp. 9876–9886.

[41] P. Chu, J. Wang, Q. You, H. Ling, and Z. Liu, "Spatial-temporal graph transformer for multiple object tracking," *arXiv e-prints*, pp. arXiv–2104, 2021.

[42] T. Meinhardt, A. Kirillov, L. Leal-Taixe, and C. Feichtenhofer, "Trackformer: Multi-object tracking with transformers," *arXiv preprint arXiv:2101.02702*, 2021.

[43] F. Zeng, B. Dong, T. Wang, C. Chen, X. Zhang, and Y. Wei, "Motr: End-to-end multiple-object tracking with transformer," *arXiv preprint arXiv:2105.03247*, 2021.

[44] Y. Xu, X. Zhou, S. Chen, and F. Li, "Deep learning for multiple object tracking: a survey," *IET CV*, vol. 13, no. 4, pp. 355–368, 2019.

[45] P. Emami, P. M. Pardalos, L. Elefteriadou, and S. Ranka, "Machine learning methods for data association in multi-object tracking," *ACM CSUR*, vol. 53, no. 4, pp. 1–34, 2020.

[46] G. Ciaparrone, F. L. Sánchez, S. Tabik, L. Troiano, R. Tagliaferri, and F. Herrera, "Deep learning in video multi-object tracking: A survey," *Neurocomputing*, vol. 381, pp. 61–88, 2020.

[47] Y. Park, L. M. Dang, S. Lee, D. Han, and H. Moon, "Multiple object tracking in deep learning approaches: A survey," *Electronics*, vol. 10, no. 19, p. 2406, 2021.

[48] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, and T.-K. Kim, "Multiple object tracking: A literature review," *AI*, vol. 293, p. 103448, 2021.

[49] S. M. Marvasti-Zadeh, L. Cheng, H. Ghanei-Yakhdan, and S. Kasaei, "Deep learning for visual tracking: A comprehensive survey," *IEEE TITS*, 2021.

[50] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *NeurIPS*, vol. 25, pp. 1097–1105, 2012.

[51] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *arXiv preprint arXiv:1405.3531*, 2014.

[52] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[53] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *CVPR*, 2015, pp. 1–9.

[54] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[55] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *IJCV*, vol. 115, no. 3, pp. 211–252, 2015.

[56] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. ˇCehovin Zajc, T. Vojir, G. Bhat, A. Lukezic, A. Eldesokey *et al.*, "The sixth visual object tracking vot2018 challenge results," in *ECCV Workshops*, 2018, pp. 0–0.

[57] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, R. Pflugfelder, J.-K. Kamarainen, L. ˇCehovin Zajc, O. Drbohlav, A. Lukezic, A. Berg *et al.*, "The seventh visual object tracking vot2019 challenge results," in *ICCV Workshops*, 2019, pp. 0–0.

[58] Z. Zhu, Q. Wang, B. Li, W. Wu, J. Yan, and W. Hu, "Distractor-aware siamese networks for visual object tracking," in *ECCV*, 2018, pp. 101–117.

[59] X. Li, C. Ma, B. Wu, Z. He, and M.-H. Yang, "Target-aware deep tracking," in *CVPR*, 2019, pp. 1369–1378.

[60] H. Morimitsu, "Multiple context features in siamese networks for visual object tracking," in *ECCV Workshops*, 2018, pp. 0–0.

[61] J. Gao, T. Zhang, and C. Xu, "Graph convolutional tracking," in *CVPR*, 2019, pp. 4649–4659.

[62] Z. Zhu, W. Wu, W. Zou, and J. Yan, "End-to-end flow correlation tracking with spatial-temporal attention," in *CVPR*, 2018, pp. 548–557.

[63] Y. Chen, L. Jing, E. Vahdani, L. Zhang, M. He, and Y. Tian, "Multi-camera vehicle tracking and re-identification on ai city challenge 2019." in *CVPR Workshops*, vol. 2, 2019.

[64] M. Cen and C. Jung, "Fully convolutional siamese fusion networks for object tracking," in *ICIP*, 2018, pp. 3718–3722.

[65] H. Fan and H. Ling, "Siamese cascaded region proposal networks for real-time visual tracking," in *CVPR*, 2019, pp. 7952–7961.

[66] F. Du, P. Liu, W. Zhao, and X. Tang, "Correlation-guided attention for corner detection based visual tracking," in *CVPR*, 2020, pp. 6836–6845.

[67] Y. Yu, Y. Xiong, W. Huang, and M. R. Scott, "Deformable siamese attention networks for visual object tracking," in *CVPR*, 2020, pp. 6728–6737.

[68] S. Karthik, A. Prabhu, and V. Gandhi, "Simple unsupervised multi-object tracking," *arXiv preprint arXiv:2006.02609*, 2020.

[69] Y. Zhang, H. Sheng, Y. Wu, S. Wang, W. Ke, and Z. Xiong, "Multiplex labeling graph for near-online tracking in crowded scenes," *IEEE IOT Journal*, vol. 7, no. 9, pp. 7892–7902, 2020.

[70] N. L. Baisa, "Occlusion-robust online multi-object visual tracking using a gm-phd filter with cnn-based re-identification," *JVCIR*, vol. 80, p. 103279, 2021.

[71] F. Yang, X. Chang, S. Sakti, Y. Wu, and S. Nakamura, "Remot: A model-agnostic refinement for multiple object tracking," *IVC*, vol. 106, p. 104091, 2021.

[72] W. Feng, Z. Hu, W. Wu, J. Yan, and W. Ouyang, "Multi-object tracking with multiple cues and switcher-aware classification," *arXiv preprint arXiv:1901.06129*, 2019.

[73] P. Chu and H. Ling, "Famnet: Joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking," in *ICCV*, 2019, pp. 6172–6181.

[74] J. Yin, W. Wang, Q. Meng, R. Yang, and J. Shen, "A unified object motion and affinity model for online multi-object tracking," in *CVPR*, 2020, pp. 6768–6777.

[75] B. Shuai, A. Berneshawi, X. Li, D. Modolo, and J. Tighe, "Siammot: Siamese multi-object tracking," in *CVPR*, 2021, pp. 12 372–12 382.

[76] C. Liang, Z. Zhang, X. Zhou, B. Li, Y. Lu, and W. Hu, "One more check: Making" fake background" be tracked again," *arXiv preprint arXiv:2104.09441*, 2021.

[77] X. Lin, Y.-a. Guo, and J. Wang, "Global correlation network: End-to-end joint multi-object detection and tracking," *arXiv preprint arXiv:2103.12511*, 2021.

[78] J. Pang, L. Qiu, X. Li, H. Chen, Q. Li, T. Darrell, and F. Yu, "Quasi-dense similarity learning for multiple object tracking," in *CVPR*, 2021, pp. 164–173.

[79] L. Jiao, R. Zhang, F. Liu, S. Yang, B. Hou, L. Li, and X. Tang, "New generation deep learning for video object detection: A survey," *IEEE TNNLS*, 2021.

[80] C. Guo, B. Fan, J. Gu, Q. Zhang, S. Xiang, V. Prinet, and C. Pan, "Progressive sparse local attention for video object detection," in *ICCV*, 2019, pp. 3909–3918.

[81] F. Xiao and Y. J. Lee, "Video object detection with an aligned spatial-temporal memory," in *ECCV*, 2018, pp. 485–501.

[82] Y. Chai, "Patchwork: A patch-wise attention network for efficient object detection and segmentation in video streams," in *ICCV*, 2019, pp. 3415–3424.

[83] G. Bertasius, L. Torresani, and J. Shi, "Object detection in video with spatiotemporal sampling networks," in *ECCV*, 2018, pp. 331–346.

[84] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Detect to track and track to detect," in *ICCV*, 2017, pp. 3038–3046.

[85] M. Ye, J. Shen, G. Lin, T. Xiang, L. Shao, and S. C. Hoi, "Deep learning for person re-identification: A survey and outlook," *IEEE TPAMI*, 2021.

[86] Y.-C. Chen, X. Zhu, W.-S. Zheng, and J.-H. Lai, "Person re-identification by camera correlation aware feature augmentation," *IEEE TPAMI*, vol. 40, no. 2, pp. 392–408, 2017.

[87] L. Zheng, Y. Yang, and A. G. Hauptmann, "Person re-identification: Past, present and future," *arXiv preprint arXiv:1610.02984*, 2016.

[88] L. Zheng, H. Zhang, S. Sun, M. Chandraker, Y. Yang, and Q. Tian, "Person re-identification in the wild," in *CVPR*, 2017, pp. 1367–1376.

[89] L. Zhao, X. Li, Y. Zhuang, and J. Wang, "Deeply-learned part-aligned representations for person re-identification," in *ICCV*, 2017, pp. 3219–3228.

[90] H. Yao, S. Zhang, R. Hong, Y. Zhang, C. Xu, and Q. Tian, "Deep representation learning with part loss for person re-identification," *IEEE TIP*, vol. 28, no. 6, pp. 2860–2871, 2019.

[91] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, "Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline)," in *ECCV*, 2018, pp. 480–496.

[92] C. Su, S. Zhang, J. Xing, W. Gao, and Q. Tian, "Deep attributes driven multi-camera person re-identification," in *ECCV*, 2016, pp. 475–491.

[93] Y. Lin, L. Zheng, Z. Zheng, Y. Wu, Z. Hu, C. Yan, and Y. Yang, "Improving person re-identification by attribute and identity learning," *Pattern Recognition*, vol. 95, pp. 151–161, 2019.

[94] T. Matsukawa and E. Suzuki, "Person re-identification using cnn features learned from combination of attributes," in *ICPR*, 2016, pp. 2428–2433.

[95] T. Wang, S. Gong, X. Zhu, and S. Wang, "Person re-identification by video ranking," in *ECCV*, 2014, pp. 688–703.

[96] K. Liu, B. Ma, W. Zhang, and R. Huang, "A spatio-temporal appearance representation for viceo-based pedestrian re-identification," in *ICCV*, 2015, pp. 3810–3818.

[97] J. Dai, P. Zhang, D. Wang, H. Lu, and H. Wang, "Video person re-identification by temporal residual learning," *IEEE TIP*, vol. 28, no. 3, pp. 1366–1377, 2018.

[98] Y. Ye, X. Ke, and Z. Yu, "A cost matrix optimization method based on spatial constraints under hungarian algorithm," in *ICRAI*, 2020, pp. 134–139.

[99] N. L. Baisa, "Online multi-object visual tracking using a gm-phd filter with deep appearance learning," in *FUSION*, 2019, pp. 1–8.

[100] Y.-c. Yoon, A. Boragule, Y.-m. Song, K. Yoon, and M. Jeon, "Online multi-object tracking with historical appearance matching and scene adaptive detection filtering," in *AVSS*, 2018, pp. 1–6.

[101] M. Babaee, Z. Li, and G. Rigoll, "A dual cnn–rnn for multiple people tracking," *Neurocomputing*, vol. 368, pp. 69–83, 2019.

[102] J. Chen, H. Sheng, Y. Zhang, and Z. Xiong, "Enhancing detection model for multiple hypothesis tracking," in *CVPR Workshops*, 2017, pp. 18–27.

[103] L. Chen, H. Ai, Z. Zhuang, and C. Shang, "Real-time multiple people tracking with deeply learned candidate selection and person re-identification," in *ICME*, 2018, pp. 1–6.

[104] E. Ristani and C. Tomasi, "Features for multi-target multi-camera tracking and re-identification," in *CVPR*, 2018, pp. 6036–6046.

[105] H. Shen, L. Huang, C. Huang, and W. Xu, "Tracklet association tracker: An end-to-end learning-based association approach for multi-object tracking," *arXiv preprint arXiv:1808.01562*, 2018.

[106] W. Li, Y. Xiong, S. Yang, M. Xu, Y. Wang, and W. Xia, "Semi-tcl: Semi-supervised track contrastive representation learning," *arXiv preprint arXiv:2107.02396*, 2021.

[107] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *CVPR*, 2015, pp. 815–823.

[108] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *NeurIPS*, vol. 33, 2020.

[109] N. L. Baisa, "Robust online multi-target visual tracking using a hisp filter with discriminative deep appearance learning," *JVCIR*, vol. 77, p. 102952, 2021.

[110] D. Neven, B. D. Brabandere, M. Proesmans, and L. V. Gool, "Instance segmentation by jointly optimizing spatial embeddings and clustering bandwidth," in *CVPR*, 2019, pp. 8837–8845.

[111] Z. Xu, W. Zhang, X. Tan, W. Yang, H. Huang, S. Wen, E. Ding, and L. Huang, "Segment as points for efficient online multi-object tracking and segmentation," in *ECCV*, 2020, pp. 264–281.

[112] Z. Xu, W. Zhang, X. Tan, W. Yang, X. Su, Y. Yuan, H. Zhang, S. Wen, E. Ding, and L. Huang, "Pointtrack++ for effective

online multi-object tracking and segmentation," *arXiv preprint arXiv:2007.01549*, 2020.

[113] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *ICIP*, 2017, pp. 3645–3649.

[114] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016.

[115] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *ICCV*, 2015, pp. 1116–1124.

[116] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *CVPR*, 2014, pp. 152–159.

[117] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *ECCV*, 2016, pp. 17–35.

[118] D. Gray, S. Brennan, and H. Tao, "Evaluating appearance models for recognition, reacquisition, and tracking," in *PETS*, vol. 3, no. 5, 2007, pp. 1–7.

[119] B. J. Prosser, W.-S. Zheng, S. Gong, T. Xiang, Q. Mary *et al.*, "Person re-identification by support vector ranking." in *BMVC*, vol. 2, no. 5, 2010, p. 6.

[120] L. Wei, S. Zhang, W. Gao, and Q. Tian, "Person transfer gan to bridge domain gap for person re-identification," in *CVPR*, 2018, pp. 79–88.

[121] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *ICIP*, 2016, pp. 3464–3468.

[122] A. Li, M. Thotakuri, D. A. Ross, J. Carreira, A. Vostrikov, and A. Zisserman, "The ava-kinetics localized human actions video dataset," *arXiv preprint arXiv:2005.00214*, 2020.

[123] K. Corona, K. Osterdahl, R. Collins, and A. Hoogs, "Meva: A large-scale multiview, multimodal video dataset for activity detection," in *WACV*, 2021, pp. 1060–1068.

[124] S. Manen, M. Gygli, D. Dai, and L. Van Gool, "Pathtrack: Fast trajectory annotation with path supervision," in *ICCV*, 2017, pp. 290–299.

[125] Q. Liu, B. Liu, Y. Wu, W. Li, and N. Yu, "Real-time online multi-object tracking in compressed domain," *IEEE Access*, vol. 7, pp. 76 489–76 499, 2019.

[126] L. Leal-Taixé, C. Canton-Ferrer, and K. Schindler, "Learning by tracking: Siamese cnn for robust target association," in *CVPR Workshops*, 2016, pp. 33–40.

[127] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Scandinavian conference on Image analysis*, 2003, pp. 363–370.

[128] E. Baser, V. Balasubramanian, P. Bhattacharyya, and K. Czarnecki, "Fantrack: 3d multi-object tracking with feature association network," in *IV*, 2019, pp. 1426–1433.

[129] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *NeurIPS*, vol. 28, pp. 91–99, 2015.

[130] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *ICCV*, 2017, pp. 2980–2988.

[131] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[132] Q. Liu, D. Chen, Q. Chu, L. Yuan, B. Liu, L. Zhang, and N. Yu, "Online multi-object tracking with unsupervised re-identification learning and occlusion estimation," *arXiv preprint arXiv:2201.01297*, 2022.

[133] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," *arXiv preprint arXiv:1904.07850*, 2019.

[134] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in *CVPR*, 2018, pp. 2403–2412.

[135] T. Yin, X. Zhou, and P. Krahenbuhl, "Center-based 3d object detection and tracking," in *CVPR*, 2021, pp. 11 784–11 793.

[136] Z. Xu, A. Meng, Z. Shi, W. Yang, Z. Chen, and L. Huang, "Continuous copy-paste for one-stage multi-object tracking and segmentation," in *ICCV*, October 2021, pp. 15 323–15 332.

[137] Y. Xu, A. Osep, Y. Ban, R. Horaud, L. Leal-Taixé, and X. Alameda-Pineda, "How to train your deep multi-object tracker," in *CVPR*, 2020, pp. 6787–6796.

[138] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014, pp. 740–755.

[139] A. Ess, B. Leibe, K. Schindler, and L. Van Gool, "A mobile vision system for robust multi-person tracking," in *CVPR*, 2008, pp. 1–8.

[140] S. Zhang, R. Benenson, and B. Schiele, "Citypersons: A diverse dataset for pedestrian detection," in *CVPR*, 2017, pp. 3213–3221.

[141] S. Shao, Z. Zhao, B. Li, T. Xiao, G. Yu, X. Zhang, and J. Sun, "Crowdhuman: A benchmark for detecting human in a crowd," *arXiv preprint arXiv:1805.00123*, 2018.

[142] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *CVPR*, 2009, pp. 304–311.

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

[144] P. Dendorfer, H. Rezatofighi, A. Milan, J. Shi, D. Cremers, I. Reid, S. Roth, K. Schindler, and L. Leal-Taixé, "Mot20: A benchmark for multi object tracking in crowded scenes," *arXiv preprint arXiv:2003.09003*, 2020.

[145] T. Xiao, S. Li, B. Wang, L. Lin, and X. Wang, "Joint detection and identification feature learning for person search," in *CVPR*, 2017, pp. 3415–3424.

[146] S. Sun, N. Akhtar, H. Song, A. Mian, and M. Shah, "Deep affinity network for multiple object tracking," *IEEE TPAMI*, vol. 43, no. 1, pp. 104–119, 2019.

[147] Y. Xu, Y. Ban, G. Delorme, C. Gan, D. Rus, and X. Alameda-Pineda, "Transcenter: Transformers with dense queries for multiple-object tracking," *arXiv preprint arXiv:2103.15145*, 2021.

[148] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?" in *CVPR*, 2018, pp. 6546–6555.

[149] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *CVPR*, 2019, pp. 658–666.

[150] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017, pp. 2961–2969.

[151] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *NeurIPS*, 2015, pp. 802–810.

[152] M. Liu and M. Zhu, "Mobile video object detection with temporally-aware feature maps," in *CVPR*, 2018, pp. 5686–5695.

[153] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. H. Torr, "Deeply supervised salient object detection with short connections," in *CVPR*, 2017, pp. 3203–3212.

[154] J. Wu, J. Cao, L. Song, Y. Wang, M. Yang, and J. Yuan, "Track to detect and segment: An online multi-object tracker," in *CVPR*, 2021, pp. 12 352–12 361.

[155] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *ICCV*, 2017, pp. 764–773.

[156] P. Li, J. Shi, and S. Shen, "Joint spatial-temporal optimization for stereo 3d object tracking," in *CVPR*, June 2020.

[157] S. Jin, W. Liu, W. Ouyang, and C. Qian, "Multi-person articulated tracking with spatial and temporal embeddings," in *CVPR*, June 2019.

[158] W. Ruan, W. Liu, Q. Bao, J. Chen, Y. Cheng, and T. Mei, "Poinet: pose-guided ovonic insight network for multi-person pose tracking," in *ACM MM*, 2019, pp. 284–292.

[159] X. Chen, S. M. Iranmanesh, and K.-C. Lien, "Patchtrack: Multiple object tracking using frame patches," *arXiv preprint arXiv:2201.00080*, 2022.

[160] G. Welch, G. Bishop *et al.*, "An introduction to the kalman filter," 1995.

[161] C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan, and M. Grohe, "Weisfeiler and leman go neural: Higher-order graph neural networks," in *AAAI*, vol. 33, no. 01, 2019, pp. 4602–4609.

[162] J. Zhang, S. Zhou, X. Chang, F. Wan, J. Wang, Y. Wu, and D. Huang, "Multiple object tracking by flowing and fusing," *arXiv preprint arXiv:2001.11180*, 2020.

[163] Q. Chu, W. Ouyang, B. Liu, F. Zhu, and N. Yu, "Dasot: A unified framework integrating data association and single object tracking for online multi-object tracking," in *AAAI*, vol. 34, no. 07, 2020, pp. 10 672–10 679.

[164] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *CVPR*, 2017, pp. 2117–2125.

[165] Q. Wang, Y. Zheng, P. Pan, and Y. Xu, "Multiple object tracking with correlation learning," in *CVPR*, 2021, pp. 3876–3886.

[166] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional siamese networks for object tracking," in *ECCV*, 2016, pp. 850–865.

[167] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with siamese region proposal network," in *CVPR*, 2018, pp. 8971–8980.

[168] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *ECCV*, 2016, pp. 21–37.

[169] W. Li, Y. Xiong, S. Yang, S. Deng, and W. Xia, "Smot: Single-shot multi object tracking," *arXiv preprint arXiv:2010.16031*, 2020.

[170] P. Sun, Y. Jiang, R. Zhang, E. Xie, J. Cao, X. Hu, T. Kong, Z. Yuan, C. Wang, and P. Luo, "Transtrack: Multiple-object tracking with transformer," *arXiv preprint arXiv:2012.15460*, 2020.

[171] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017, pp. 5998–6008.

[172] L. Zheng, M. Tang, Y. Chen, G. Zhu, J. Wang, and H. Lu, "Improving multiple object tracking with single object tracking," in *CVPR*, 2021, pp. 2453–2462.

[173] A. Girbau, X. Giró-i Nieto, I. Rius, and F. Marqués, "Multiple object tracking with mixture density networks for trajectory estimation," *arXiv preprint arXiv:2106.10950*, 2021.

[174] X. Jiang, P. Li, Y. Li, and X. Zhen, "Graph neural based end-to-end data association framework for online multiple-object tracking," *arXiv preprint arXiv:1907.05315*, 2019.

[175] S. Guo, J. Wang, X. Wang, and D. Tao, "Online multiple object tracking with cross-task synergy," in *CVPR*, 2021, pp. 8136–8145.

[176] S. Wang, H. Sheng, Y. Zhang, Y. Wu, and Z. Xiong, "A general recurrent tracking framework without real data," in *ICCV*, 2021, pp. 13 219–13 228.

[177] P. Tokmakov, J. Li, W. Burgard, and A. Gaidon, "Learning to track with object permanence," *arXiv preprint arXiv:2103.14258*, 2021.

[178] N. Ballas, L. Yao, C. Pal, and A. Courville, "Delving deeper into convolutional networks for learning video representations," *arXiv preprint arXiv:1511.06432*, 2015.

[179] C. Kim, L. Fuxin, M. Alotaibi, and J. M. Rehg, "Discriminative appearance modeling with multi-track pooling for real-time multi-object tracking," in *CVPR*, 2021, pp. 9553–9562.

[180] C. Kim, F. Li, and J. M. Rehg, "Multi-object tracking with neural gating using bilinear lstm," in *ECCV*, 2018, pp. 200–215.

[181] F. Saleh, S. Aliakbarian, H. Rezatofighi, M. Salzmann, and S. Gould, "Probabilistic tracklet scoring and inpainting for multiple object tracking," in *CVPR*, 2021, pp. 14 329–14 339.

[182] G. Wang, Y. Wang, R. Gu, W. Hu, and J.-N. Hwang, "Split and connect: A universal tracklet booster for multi-object tracking," *IEEE TMM*, 2022.

[183] J. Peng, T. Wang, W. Lin, J. Wang, J. See, S. Wen, and E. Ding, "Tpm: Multiple object tracking with tracklet-plane matching," *Pattern Recognition*, vol. 107, p. 107480, 2020.

[184] H.-M. Hsu, T.-W. Huang, G. Wang, J. Cai, Z. Lei, and J.-N. Hwang, "Multi-camera tracking of vehicles based on deep features re-id and trajectory-based camera link models." in *CVPR Workshops*, 2019, pp. 416–424.

[185] J. Gao and R. Nevatia, "Revisiting temporal modeling for video-based person reid," *arXiv preprint arXiv:1805.02104*, 2018.

[186] P. Dai, R. Weng, W. Choi, C. Zhang, Z. He, and W. Ding, "Learning a proposal classifier for multiple object tracking," in *CVPR*, 2021, pp. 2443–2452.

[187] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[188] J. Xiang, G. Xu, C. Ma, and J. Hou, "End-to-end learning deep crf models for multi-object tracking deep crf models," *IEEE TCSVT*, vol. 31, no. 1, pp. 275–288, 2020.

[189] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, "Conditional random fields as recurrent neural networks," in *ICCV*, 2015, pp. 1529–1537.

[190] M. Larsson, F. Kahl, S. Zheng, A. Arnab, P. H. Torr, and R. I. Hartley, "Learning arbitrary potentials in crfs with gradient descent." 2017.

[191] M. Babaee, A. Athar, and G. Rigoll, "Multiple people tracking using hierarchical deep tracklet re-identification," *arXiv preprint arXiv:1811.04091*, 2018.

[192] D. Reid, "An algorithm for tracking multiple targets," *IEEE TAC*, vol. 24, no. 6, pp. 843–854, 1979.

[193] Y. Zhang, H. Sheng, Y. Wu, S. Wang, W. Lyu, W. Ke, and Z. Xiong, "Long-term tracking with deep tracklet association," *IEEE TIP*, vol. 29, pp. 6694–6706, 2020.

[194] G. Wang, Y. Wang, H. Zhang, R. Gu, and J.-N. Hwang, "Exploit the connectivity: Multi-object tracking with trackletnet," in *ACM MM*, 2019, pp. 482–490.

[195] Y.-C. Yoon, D. Y. Kim, Y.-m. Song, K. Yoon, and M. Jeon, "Online multiple pedestrians tracking using deep temporal appearance matching association," *Information Sciences*, vol. 561, pp. 326–351, 2021.

[196] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei, "Relation networks for object detection," in *CVPR*, 2018, pp. 3588–3597.

[197] J. Xu, Y. Cao, Z. Zhang, and H. Hu, "Spatial-temporal relation networks for multi-object tracking," in *ICCV*, 2019, pp. 3988–3998.

[198] F. Bastani, S. He, and S. Madden, "Self-supervised multi-object tracking with cross-input consistency," *NeurIPS*, vol. 34, 2021.

[199] J. Zhu, H. Yang, N. Liu, M. Kim, W. Zhang, and M.-H. Yang, "Online multi-object tracking with dual matching attention networks," in *ECCV*, 2018, pp. 366–382.

[200] W.-C. Hung, H. Kretzschmar, T.-Y. Lin, Y. Chai, R. Yu, M.-H. Yang, and D. Anguelov, "Soda: Multi-object tracking with soft data association," *arXiv preprint arXiv:2008.07725*, 2020.

[201] Q. Liu, Q. Chu, B. Liu, and N. Yu, "Gsm: Graph similarity model for multi-object tracking." in *IJCAI*, 2020, pp. 530–536.

[202] I. Papakis, A. Sarkar, and A. Karpatne, "Gcnnmatch: Graph convolutional neural networks for multi-object tracking via sinkhorn normalization," *arXiv preprint arXiv:2010.00067*, 2020.

[203] J. Li, X. Gao, and T. Jiang, "Graph networks for multiple object tracking," in *WACV*, 2020, pp. 719–728.

[204] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *ICML*, 2017, pp. 1263–1272.

[205] M. Guo, E. Chou, D.-A. Huang, S. Song, S. Yeung, and L. Fei-Fei, "Neural graph matching networks for fewshot 3d action recognition," in *ECCV*, 2018, pp. 653–669.

[206] P. W. Battaglia, R. Pascanu, M. Lai, D. Rezende, and K. Kavukcuoglu, "Interaction networks for learning about objects, relations and physics," *arXiv preprint arXiv:1612.00222*, 2016.

[207] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner *et al.*, "Relational inductive biases, deep learning, and graph networks," *arXiv preprint arXiv:1806.01261*, 2018.

[208] A. Rangesh, P. Maheshwari, M. Gebre, S. Mhatre, V. Ramezani, and M. M. Trivedi, "Trackmpnn: A message passing graph neural architecture for multi-object tracking," *arXiv preprint arXiv:2101.04206*, 2021.

[209] C. Shan, C. Wei, B. Deng, J. Huang, X.-S. Hua, X. Cheng, and K. Liang, "Tracklets predicting based adaptive graph tracking," *arXiv preprint arXiv:2010.09015*, 2020.

[210] J. He, Z. Huang, N. Wang, and Z. Zhang, "Learnable graph matching: Incorporating graph partitioning with deep feature learning for multiple object tracking," in *CVPR*, 2021, pp. 5299–5309.

[211] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*, 2012, pp. 3354–3361.

[212] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *IJRR*, vol. 32, no. 11, pp. 1231–1237, 2013.

[213] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "Motchallenge 2015: Towards a benchmark for multi-target tracking," *arXiv preprint arXiv:1504.01942*, 2015.

[214] L. Wen, D. Du, Z. Cai, Z. Lei, M.-C. Chang, H. Qi, J. Lim, M.-H. Yang, and S. Lyu, "Ua-detrac: A new benchmark and protocol for multi-object detection and tracking," *CVIU*, vol. 193, p. 102907, 2020.

[215] M. Andriluka, U. Iqbal, E. Insafutdinov, L. Pishchulin, A. Milan, J. Gall, and B. Schiele, "Posetrack: A benchmark for human pose estimation and tracking," in *CVPR*, 2018, pp. 5167–5176.

[216] U. Iqbal, A. Milan, and J. Gall, "Posetrack: Joint multi-person pose estimation and tracking," in *CVPR*, 2017, pp. 2011–2020.

[217] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *CVPR*, 2020, pp. 11 621–11 631.

[218] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *CVPR*, 2020, pp. 2446–2454.

[219] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," in *CVPR*, 2020, pp. 2636–2645.

[220] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving video database with scalable annotation tooling," *arXiv preprint arXiv:1805.04687*, vol. 2, no. 5, p. 6, 2018.

[221] P. Zhu, L. Wen, D. Du, X. Bian, H. Fan, Q. Hu, and H. Ling, "Detection and tracking meet drones challenge," *IEEE TPAMI*, no. 01, pp. 1–1, 2021.

[222] H. Fan, D. Du, L. Wen, P. Zhu, Q. Hu, H. Ling, M. Shah, J. Pan, A. Schumann, B. Dong *et al.*, "Visdrone-mot2020: The vision meets drone multiple object tracking challenge results," in *ECCV*, 2020, pp. 713–727.

[223] L. Wen, P. Zhu, D. Du, X. Bian, H. Ling, Q. Hu, J. Zheng, T. Peng, X. Wang, Y. Zhang *et al.*, "Visdrone-mot2019: The vision meets drone multiple object tracking challenge results," in *ICCV Workshops*, 2019, pp. 0–0.

[224] E. Bochinski, T. Senst, and T. Sikora, "Extending iou based multi-object tracking by visual information," in *AVSS*, 2018, pp. 1–6.

[225] W. Lin, H. Liu, S. Liu, Y. Li, R. Qian, T. Wang, N. Xu, H. Xiong, G.-J. Qi, and N. Sebe, "Human in events: A large-scale benchmark for human-centric video analysis in complex events," *arXiv preprint arXiv:2005.04490*, 2020.

[226] L. Xu, R. Xu, and S. Jin, "Hieve acm mm grand challenge 2020: Pose tracking in crowded scenes," in *ACM MM*, 2020, pp. 4689–4693.

[227] P. Sun, J. Cao, Y. Jiang, Z. Yuan, S. Bai, K. Kitani, and P. Luo, "Dancetrack: Multi-object tracking in uniform appearance and diverse motion," *arXiv preprint arXiv:2111.14690*, 2021.

[228] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Conference on robot learning*, 2017, pp. 1–16.

[229] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, "Virtual worlds as proxy for multi-object tracking analysis," in *CVPR*, 2016, pp. 4340–4349.

[230] A. Dave, T. Khurana, P. Tokmakov, C. Schmid, and D. Ramanan, "Tao: A large-scale benchmark for tracking any object," in *ECCV*, 2020, pp. 436–454.

[231] T. Chavdarova, P. Baqué, S. Bouquet, A. Maksai, C. Jose, T. Bagautdinov, L. Lettry, P. Fua, L. Van Gool, and F. Fleuret, "Wildtrack: A multi-camera hd dataset for dense unscripted pedestrian detection," in *CVPR*, 2018, pp. 5030–5039.

[232] H. Bai, W. Cheng, P. Chu, J. Liu, K. Zhang, and H. Ling, "Gmot-40: A benchmark for generic multiple object tracking," in *CVPR*, 2021, pp. 6719–6728.

[233] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: the clear mot metrics," *EURASIP Journal on Image and Video Processing*, vol. 2008, pp. 1–10, 2008.

[234] J. Luiten, A. Osep, P. Dendorfer, P. Torr, A. Geiger, L. Leal-Taixé, and B. Leibe, "Hota: A higher order metric for evaluating multi-object tracking," *IJCV*, vol. 129, no. 2, pp. 548–578, 2021.

[235] W. Feng, Z. Hu, B. Li, W. Gan, W. Wu, and W. Ouyang, "Samot: Switcher-aware multi-object tracking and still another mot measure," *arXiv preprint arXiv:2009.10338*, 2020.

[236] J. Valmadre, A. Bewley, J. Huang, C. Sun, C. Sminchisescu, and C. Schmid, "Local metrics for multi-object tracking," *arXiv preprint arXiv:2104.02631*, 2021.

[237] A. Shenoi, M. Patel, J. Gwak, P. Goebel, A. Sadeghian, H. Rezatofighi, R. Martín-Martín, and S. Savarese, "Jrmot: A real-time 3d multi-object tracker and a new large-scale dataset," in *IROS*, 2020, pp. 10 335–10 342.

[238] D. Mykheievskyi, D. Borysenko, and V. Porokhonskyy, "Learning local feature descriptors for multiple object tracking," in *ACCV*, 2020.

[239] W. Zhang, H. Zhou, S. Sun, Z. Wang, J. Shi, and C. C. Loy, "Robust multi-modality multi-object tracking," in *ICCV*, 2019, pp. 2365–2374.

[240] S. Wang, Y. Sun, C. Liu, and M. Liu, "Pointtracknet: An end-to-end network for 3-d object detection and tracking from point clouds," *IEEE RAL*, vol. 5, no. 2, pp. 3206–3212, 2020.

[241] H. Wu, Q. Li, C. Wen, X. Li, X. Fan, and C. Wang, "Tracklet proposal network for multi-object tracking on point clouds," in *IJCAI*, 2021, pp. 1165–1171.

[242] S. Wang, P. Cai, L. Wang, and M. Liu, "Ditnet: End-to-end 3d object detection and track id assignment in spatio-temporal world," *IEEE RAL*, vol. 6, no. 2, pp. 3397–3404, 2021.

[243] K. Huang and Q. Hao, "Joint multi-object detection and tracking with camera-lidar fusion for autonomous driving," in *IROS*, 2021, pp. 6983–6989.

[244] H.-N. Hu, Y.-H. Yang, T. Fischer, T. Darrell, F. Yu, and M. Sun, "Monocular quasi-dense 3d object tracking," *arXiv preprint arXiv:2103.07351*, 2021.

[245] I. Ruiz, L. Porzi, S. R. Bulò, P. Kontschieder, and J. Serrat, "Weakly supervised multi-object tracking and segmentation." in *WACV Workshops*, 2021, pp. 125–133.

[246] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *ICCV*, 2017, pp. 618–626.

[247] D. McKee, B. Shuai, A. Berneshawi, M. Wang, D. Modolo, S. Lazebnik, and J. Tighe, "Multi-object tracking with hallucinated and unlabeled videos," *arXiv preprint arXiv:2108.08836*, 2021.

[248] K. Nishimura, J. Hayashida, C. Wang, R. Bise *et al.*, "Weakly-supervised cell tracking via backward-and-forward propagation," in *ECCV*, 2020, pp. 104–121.

[249] A. Wu, C. Lin, B. Chen, W. Huang, Z. Huang, and W.-S. Zheng, "Transductive multi-object tracking in complex events by interactive self-training," in *ACM MM*, 2020, pp. 4620–4624.

[250] S. Yoon, K. Shim, K. Park, and C. Kim, "Weakly-supervised multiple object tracking via a masked center point warping loss," in *ICIP*, 2021, pp. 1164–1168.

[251] Z. He, J. Li, D. Liu, H. He, and D. Barber, "Tracking by animation: Unsupervised learning of multi-object attentive trackers," in *CVPR*, 2019, pp. 1318–1327.

[252] C.-H. H. Yang, M. Chhabra, Y.-C. Liu, Q. Kong, T. Yoshinaga, and T. Murakam, "Robust unsupervised multi-object tracking in noisy environments," *arXiv preprint arXiv:2105.10005*, 2021.

[253] Y. Liu, Z. Wang, X. Zhou, and L. Zheng, "Synthetic data are as good as the real for association knowledge learning in multi-object tracking," *arXiv preprint arXiv:2106.16100*, 2021.

[254] M. Fabbri, G. Brasó, G. Maugeri, O. Cetintas, R. Gasparini, A. Ošep, S. Calderara, L. Leal-Taixe, and R. Cucchiara, "Motsynth: How can synthetic data help pedestrian detection and tracking?" in *ICCV*, 2021, pp. 10 849–10 859.

[255] A. Hornakova, R. Henschel, B. Rosenhahn, and P. Swoboda, "Lifted disjoint paths with application in multiple object tracking," in *ICML*, 2020, pp. 4364–4375.

[256] W. Song, S. Li, T. Chang, A. Hao, Q. Zhao, and H. Qin, "Cross-view contextual relation transferred network for unsupervised vehicle tracking in drone videos," in *WACV*, 2020, pp. 1707–1716.

[257] Y. Gan, R. Han, L. Yin, W. Feng, and S. Wang, "Self-supervised multi-view multi-human association and tracking," in *ACM MM*, 2021, pp. 282–290.

[258] K. G. Quach, P. Nguyen, H. Le, T.-D. Truong, C. N. Duong, M.-T. Tran, and K. Luu, "Dyglip: A dynamic graph model with link prediction for accurate multi-camera multiple object tracking," in *CVPR*, June 2021, pp. 13 784–13 793.

[259] D. M. Nguyen, R. Henschel, B. Rosenhahn, D. Sonntag, and P. Swoboda, "Lmgp: Lifted multicut meets geometry projections for multi-camera multi-object tracking," *arXiv preprint arXiv:2111.11892*, 2021.

[260] Y. Xu, X. Liu, Y. Liu, and S.-C. Zhu, "Multi-view people tracking via hierarchical trajectory composition," in *CVPR*, 2016, pp. 4256–4265.

[261] R. Han, W. Feng, Y. Zhang, J. Zhao, and S. Wang, "Multiple human association and tracking from egocentric and complementary top views," *IEEE TPAMI*, 2021.

[262] L. Chen, H. Ai, R. Chen, Z. Zhuang, and S. Liu, "Cross-view tracking for multi-human 3d pose estimation at over 100 fps," in *CVPR*, 2020, pp. 3279–3288.

[263] R. Han, W. Feng, J. Zhao, Z. Niu, Y. Zhang, L. Wan, and S. Wang, "Complementary-view multiple human tracking," in *AAAI*, vol. 34, no. 07, 2020, pp. 10 917–10 924.

[264] Y. Xu, X. Liu, L. Qin, and S.-C. Zhu, "Cross-view people tracking by scene-centered spatio-temporal parsing," in *AAAI*, vol. 31, no. 1, 2017.

[265] C. Luo, X. Yang, and A. Yuille, "Exploring simple 3d multi-object tracking for autonomous driving," in *ICCV*, 2021, pp. 10 488–10 497.

[266] Z. Pang, Z. Li, and N. Wang, "Simpletrack: Understanding and rethinking 3d multi-object tracking," *arXiv preprint arXiv:2111.09621*, 2021.

[267] S. Shi, C. Guo, J. Yang, and H. Li, "Pv-rcnn: The top-performing lidar-only solutions for 3d detection/3d tracking/domain adaptation of waymo open dataset challenges," *arXiv preprint arXiv:2008.12599*, 2020.

[268] J. Sun, Y. Xie, S. Zhang, L. Chen, G. Zhang, H. Bao, and X. Zhou, "You don't only look once: Constructing spatial-temporal memory for integrated 3d object detection and tracking," in *ICCV*, October 2021, pp. 3185–3194.

[269] T. Linder, S. Breuers, B. Leibe, and K. O. Arras, "On multi-modal people tracking from mobile platforms in very crowded and dynamic environments," in *ICRA*, 2016, pp. 5512–5519.

[270] Y. Zhong, S. You, and U. Neumann, "Modeling cross-modal interaction in a multi-detector, multi-modal tracking framework," in *ACCV*, 2020.

[271] H.-k. Chiu, J. Li, R. Ambruş, and J. Bohg, "Probabilistic 3d multi-modal, multi-object tracking for autonomous driving," in *ICRA*, 2021, pp. 14 227–14 233.

[272] A. Kim, A. Ošep, and L. Leal-Taixé, "Eagermot: 3d multi-object tracking via sensor fusion," in *ICRA*, 2021, pp. 11 315–11 321.

[273] R. Nabati, L. Harris, and H. Qi, "Cftrack: Center-based radar and camera fusion for 3d multi-object tracking," in *IV Workshops*, 2021, pp. 243–248.

[274] F. R. Valverde, J. V. Hurtado, and A. Valada, "There is more than meets the eye: Self-supervised multi-object detection and tracking with sound by distilling multimodal knowledge," in *CVPR*, 2021, pp. 11 612–11 621.

[275] N. Scheiner, F. Kraus, F. Wei, B. Phan, F. Mannan, N. Appenrodt, W. Ritter, J. Dickmann, K. Dietmayer, B. Sick, and F. Heide, "Seeing around street corners: Non-line-of-sight detection and tracking in-the-wild using doppler radar," in *CVPR*, June 2020.

[276] H. Chen, W. Cai, F. Wu, and Q. Liu, "Vehicle-mounted far-infrared pedestrian detection using multi-object tracking," *Infrared Physics & Technology*, vol. 115, p. 103697, 2021.

[277] X. Liang, L. Lee, and E. P. Xing, "Deep variation-structured reinforcement learning for visual relationship and attribute detection," in *CVPR*, 2017, pp. 848–857.

[278] Y. Rao, J. Lu, and J. Zhou, "Attention-aware deep reinforcement learning for video face recognition," in *ICCV*, 2017, pp. 3931–3940.

[279] Q. Cao, L. Lin, Y. Shi, X. Liang, and G. Li, "Attention-aware face hallucination via deep reinforcement learning," in *CVPR*, 2017, pp. 690–698.

[280] X. Kong, B. Xin, Y. Wang, and G. Hua, "Collaborative deep reinforcement learning for joint object search," in *CVPR*, 2017, pp. 1695–1704.

[281] P. Rosello and M. J. Kochenderfer, "Multi-agent reinforcement learning for multi-object tracking," in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, 2018, pp. 1397–1404.

[282] M.-x. Jiang, C. Deng, Z.-g. Pan, L.-f. Wang, and X. Sun, "Multi-object tracking in videos based on lstm and deep reinforcement learning," *Complexity*, vol. 2018, 2018.

[283] L. Ren, J. Lu, Z. Wang, Q. Tian, and J. Zhou, "Collaborative deep reinforcement learning for multi-object tracking," in *ECCV*, 2018, pp. 586–602.

[284] M. Jiang, T. Hai, Z. Pan, H. Wang, Y. Jia, and C. Deng, "Multi-agent deep reinforcement learning for multi-object tracker," *IEEE Access*, vol. 7, pp. 32 400–32 407, 2019.

[285] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey, "Meta-learning in neural networks: A survey," *arXiv preprint arXiv:2004.05439*, 2020.

[286] J. Vanschoren, "Meta-learning: A survey," *arXiv preprint arXiv:1810.03548*, 2018.

[287] S. Thrun and L. Pratt, "Learning to learn: Introduction and overview," in *Learning to learn*. Springer, 1998, pp. 3–17.

[288] H. F. Harlow, "The formation of learning sets." *Psychological review*, vol. 56, no. 1, p. 51, 1949.

[289] C. Giraud-Carrier, R. Vilalta, and P. Brazdil, "Introduction to the special issue on meta-learning," *Machine learning*, vol. 54, no. 3, pp. 187–193, 2004.

[290] A. M. Schrier, "Learning how to learn: The significance and current status of learning set formation," *Primates*, vol. 25, no. 1, pp. 95–102, 1984.

[291] J.-M. Perez-Rua, X. Zhu, T. M. Hospedales, and T. Xiang, "Incremental few-shot object detection," in *CVPR*, 2020, pp. 13 846–13 855.

[292] B. Kang, Z. Liu, X. Wang, F. Yu, J. Feng, and T. Darrell, "Few-shot object detection via feature reweighting," in *ICCV*, 2019, pp. 8420–8429.

[293] A. Shaban, S. Bansal, Z. Liu, I. Essa, and B. Boots, "One-shot learning for semantic segmentation," in *BMVC*, 2017.

[294] N. Dong and E. P. Xing, "Few-shot semantic segmentation with prototype learning." in *BMVC*, vol. 3, no. 4, 2018.

[295] K. Rakelly, E. Shelhamer, T. Darrell, A. A. Efros, and S. Levine, "Few-shot segmentation propagation with guided networks," *arXiv preprint arXiv:1806.07373*, 2018.

[296] S. Liu, A. Davison, and E. Johns, "Self-supervised generalisation with meta auxiliary learning," *NeurIPS*, vol. 32, 2019.

[297] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[298] S. Toshniwal, H. Tang, L. Lu, and K. Livescu, "Multitask learning with low-level auxiliary tasks for encoder-decoder based speech recognition," *arXiv preprint arXiv:1704.01631*, 2017.

[299] L. Liebel and M. Körner, "Auxiliary tasks in multi-task learning," *arXiv preprint arXiv:1805.06334*, 2018.

[300] M. Jaderberg, V. Mnih, W. M. Czarnecki, T. Schaul, J. Z. Leibo, D. Silver, and K. Kavukcuoglu, "Reinforcement learning with unsupervised auxiliary tasks," *arXiv preprint arXiv:1611.05397*, 2016.

[301] J. Flynn, I. Neulander, J. Philbin, and N. Snavely, "Deepstereo: Learning to predict new views from the world's imagery," in *CVPR*, 2016, pp. 5515–5524.

[302] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *CVPR*, 2017, pp. 1851–1858.

[303] Y. Du, W. M. Czarnecki, S. M. Jayakumar, M. Farajtabar, R. Pascanu, and B. Lakshminarayanan, "Adapting auxiliary losses using gradient similarity," *arXiv preprint arXiv:1812.02224*, 2018.

[304] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, p. 2, 2021.

[305] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *ICML*, 2020, pp. 1597–1607.

[306] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *NeurIPS*, vol. 33, pp. 9912–9924, 2020.

[307] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *CVPR*, 2020, pp. 9729–9738.

[308] D. Kim, D. Cho, and I. S. Kweon, "Self-supervised video representation learning with space-time cubic puzzles," in *AAAI*, vol. 33, no. 01, 2019, pp. 8545–8552.

[309] P. Goyal, D. Mahajan, A. Gupta, and I. Misra, "Scaling and benchmarking self-supervised visual representation learning," in *ICCV*, 2019, pp. 6391–6400.

[310] T. Han, W. Xie, and A. Zisserman, "Self-supervised co-training for video representation learning," *NeurIPS*, vol. 33, pp. 5679–5690, 2020.

[311] R. Qian, T. Meng, B. Gong, M.-H. Yang, H. Wang, S. Belongie, and Y. Cui, "Spatiotemporal contrastive video representation learning," in *CVPR*, 2021, pp. 6964–6974.

[312] H. Vu, T. D. Nguyen, T. Le, W. Luo, and D. Phung, "Robust anomaly detection in videos using multilevel representations," in *AAAI*, vol. 33, no. 01, 2019, pp. 5216–5223.

[313] G. Pang, C. Yan, C. Shen, A. v. d. Hengel, and X. Bai, "Self-trained deep ordinal regression for end-to-end video anomaly detection," in *CVPR*, 2020, pp. 12 173–12 182.

[314] P. Chen, D. Huang, D. He, X. Long, R. Zeng, S. Wen, M. Tan, and C. Gan, "Rspnet: Relative speed perception for unsupervised video representation learning," in *AAAI*, vol. 1, no. 3, 2021, p. 5.