Graph-Based Data Association in Multiple Object Tracking: A Survey[*]

Despoina Touska, Konstantinos Gkountakos, Theodora Tsikrika, Konstantinos Ioannidis, Stefanos Vrochidis, and Ioannis Kompatsiaris

Information Technologies Institute, Centre for Research and Technology Hellas,
Thermi-Thessaloniki, Greece
{destousok,gountakos,theodora.tsikrika,kioannid,stefanos,ikom}@iti.gr

Abstract. In Multiple Object Tracking (MOT), data association is a key component of the tracking-by-detection paradigm and endeavors to link a set of discrete object observations across a video sequence, yielding possible trajectories. Our intention is to provide a classification of numerous graph-based works according to the way they measure object dependencies and their footprint on the graph structure they construct. In particular, methods are organized into Measurement-to-Measurement (MtM), Measurement-to-Track (MtT), and Track-to-Track (TtT). At the same time, we include recent Deep Learning (DL) implementations among traditional approaches to present the latest trends and developments in the field and offer a performance comparison. In doing so, this work serves as a foundation for future research by providing newcomers with information about the graph-based bibliography of MOT.

Keywords: Multiple object tracking · Data association · Graph optimization · Graph neural networks.

1 Introduction

Multiple Object Tracking (MOT) aims to determine and maintain the identities of all the depicted objects in a video sequence and output their trajectories. The objects of interest, namely targets, can be pedestrians, vehicles, or even subcellular structures. Due to its academic importance and practical application, MOT has received a great deal of attention recently, considering the multiple related challenges [8], as well as the growing number of applications that incorporate tracking technologies. Among its applications are autonomous driving and video surveillance, such as traffic control and activity recognition [29].

Apart from the difficulty in finding multiple objects' trajectories simultaneously, some challenges for MOT algorithms stem from the changing scales of objects due to their motion and moving cameras. Additionally, the highly resembled targets and the frequent occlusions make the targets indistinguishable

[*] This work was partially supported by the projects NESTOR (H2020-101021851), INFINITY (H2020-883293), and ODYSSEUS (H2020-101021857), funded by the European Commission.

and the maintenance of their trajectories a complicated task. The complexity is further increased when the input object observations are not precise, as they can suffer from duplicated, false, or missed detections. The MOT algorithms are also subject to inherent ambiguities caused by weather conditions, varying illuminations, or shadows. Therefore, many studies [30,38] have tried over the last decades to give a solution to the MOT problem, but since of these diverse challenges, research continues.

MOT methods have been dominated by the tracking-by-detection paradigm [30], which requires linking together a set of object detections through a process known as data association. A common and general way to formulate data association is by using graph models, as they can offer a natural way to represent the MOT problem [4]. The two key concepts behind this formulation are the graph network construction as well as its optimization framework. In this sense, the solution of data association in a graph is a set of independent paths that connects the objects' observations.

Recently, several review papers have proposed different taxonomies to categorize the MOT bibliography, with many focusing on methods that adopt the tracking-by-detection paradigm [30]. Although data association through classical graph methods has already been analyzed in MOT reviews [4,10], there is no significant reference to the impact of learning on graphs, as relevant studies [37] have been conducted over the last few years. Consequently, there is still scope for investigation and analysis considering the broad task of data association. Thereby, this survey provides a comprehensive overview of data association via graph models in order to present their utility for the topic of MOT thoroughly.

In summary, the core of this work is to categorize graph-based solutions for MOT, as graphs have been inextricably linked to the topic for decades and have shown potential when dealing with complicated scenes (e.g. crowded environments). Specifically, we include a range of methods and create a concrete classification based on the different graph formulations resulting from the different types of associating the input data. In particular, the methods are classified into Measurement-to-Measurement (MtM), Measurement-to-Track (MtT), and Track-to-Track (TtT) association, where the measurement refers to object detection and the track to object trajectories. To cover a wider range of methods, our examination includes learning-based graph models among traditional model-based ones, so as to demonstrate how Deep Learning (DL) can operate in the graph domain. The analysis is also accompanied by a theoretical explanation to provide a better understanding of the basic graph notions.

2 Related Work

There is a sufficient number of existing surveyed works on the topic of MOT, proposing different categorizations. One conventional and broad category is to classify methods as online and offline [30,38,20]. In the former category, information is leveraged from past frames up to the incoming ones, while methods leverage past and future frames in the latter. Moreover, many works [30,38,5] highlight the differences between learning-based and model-based techniques in terms of the improvements the learning can bring.

Particularly, the authors in [30] draw a timeline of pedestrians-related tracking methods, emphasizing the advancements of the tracking-by-detection paradigm and providing its general formulation. In [38], learning-based MOT methods are summarized, which are competitive and top-ranked in public benchmarks. Alongside online and offline classification, the categorization continues in terms of DL contribution to deep descriptions enhancement and the construction of end-to-end methodologies. In [9], the authors focus on the data association part of MOT, introducing the linear and multidimensional assignment formulations and reviewing learning and non-learning algorithms to determine the optimal matching.

In [5], the main parts of the multiple target tracking pipeline, including target detection, track filtering, and data association, are presented. Particularly in the task of proposed tracks evaluation, the categorization of DL methods in MtM, MtT, or feature association is based on the matching score generation. In [20], the variety of model-based solutions for MOT is reviewed according to different aspects, such as the initialization strategies, the processing mode, and the randomness of the output, making it easy for newcomers to enter the field. In [8], tracking trends, such as global optimization, regression, and learning models, are presented, comparing State-of-the-Art (SotA) methods in terms of performance, along with extended details regarding the standardized framework in the context of MOTChallenge [1] evaluation.

In [10], the authors present the different implemented methodologies for feature generation and object tracking. The functions used to provide object features, such as appearance, velocity, and location, are mainly handcrafted. At the same time, the authors separate Bayesian-based from association-based detection linking methods. Regarding data association, they also demonstrate further partitioning according to local and global optimization. In [4], a methodology-based work focusing on two types of graphs is presented, where the track and factor graph approaches are discussed. However, the above surveys are too general and do not capture the algorithms used in detail in terms of data association. Thus, this work extensively examines and categorizes the graph-based methods considering the different types of associating the input data.

3 Theoretical Background

Graphs are defined as the mathematical structures that are used to describe a set of discrete objects via nodes connected by edges. A graph can be classified as undirected when its edges can be traversed in either direction or as directed if its edges have a definite direction. In graph theory, NFs are examples of directed graphs $G = (\mathbf{N}odes, \mathbf{E}dges)$. Each edge $e(u, v)$, from node u to node v, has a capacity $c(u, v) > 0$, a flow $f(u, v)$, and a cost $a(u, v)$. NFs also have two

[1] https://motchallenge.net

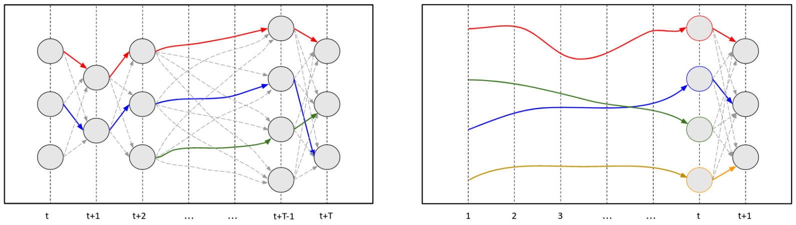
additional nodes; the source $s \in N$ and the sink $t \in N$, with no incoming and no outgoing edges, respectively. In particular, each edge's flow cannot exceed its capacity's value, and the total cost for the edge is equal to $f(u,v) \cdot a(u,v)$. Furthermore, the amount of flow that enters a node should be equal to the amount of flow that outgoes unless it is the source or the sink node.

In graph optimization, the goal of the NF problem is to find the amount of flow, or else the different paths, starting from the source node and ending at the sink node. In doing so, a linear objective function is defined with regard to the total cost of the flow over all edges, and the problem is then reduced to a minimization problem. In the case that the capacity of every edge is unit, the problem is known as the disjoint paths problem, as the resulted paths do not have any common edges.

Some types of NF problems are the Maximum Flow (MF) problem, the Minimum-Cost Flow (MCF) problem, and the assignment problem. The MF problem aims to maximize the total amount of flow in the network, while the MCF problem aims to attain a predefined (or the maximum) amount of flow with the minimum possible cost. A case of NF problems is the assignment ones, which perform in a weighted BG. In BG $G = (N, V, E)$, there are two disjoint and independent sets N and V of nodes, and a set E of edges denotes the connections between the nodes of these two sets. In case the nodes are divided into k different independent sets, the graph is called K-Partite Graph (KG).

Unlike directed graphs, where graph partitioning aims to find a set of independent paths when dealing with undirected graphs, the problem's goal is to decompose the graph nodes into mutually exclusive groups. The generalized Minimum Clique Problem (GMCP) and the Minimum Cost Multicut Problem (MCMP) are two related optimization problems in the bibliography. GMCP is applied to undirected graph $G = (\mathbf{N}odes, \mathbf{E}dges, \mathbf{w}eights)$. Given a number $k \in \mathbb{N}_{>1}$, the set of nodes is divided into k partitions of nodes that do not share common nodes. GMCP seeks to find the subgraph G_s, named clique, with the minimum total cost encompassing only one node from every partition. Each node in a clique is adjacent to each other, as there is an edge that connects them. The latter makes the subgraph G_s complete. In MCMP, graph decomposition includes the identification of a set of edges, named multicut of the graph, that straddles graph components into groups.

More recently, several studies have tried to apply deep learning to graph optimization, using Graph Neural Networks (GNNs) [26], which perform on graph-structured data. Given as input a graph with nodes and edges, GNNs attempt to discover nodes' features and learn the topology pattern of the input data by exploiting their neighboring nodes' information via feature aggregation. Different GNN models [37] have been proposed, each with a different feature aggregation function. Message Passing Neural Networks (MPNNs) -introduced in [11] - consist of multiple propagation layers, which propagate information by updating each node regarding their adjacent nodes' features. Graph Convolutional Networks (GCNs) [15] are considered a case of the message passing paradigm, with a different design in their aggregation function.



(a) NF in a T-frame batch.

(b) BP in two consecutive frames.

Fig. 1: Illustration of two basic formulations in graph-based MOT.

4 Analysis of MOT Methods

4.1 Measurement-to-Measurement Association

Most methods in this category interpret the association task as an NF problem (i.e. MCF, MF), as shown in Figure 1(a), while few perform graph partitioning as an MCMP. The association is made over a batch of frames or an entire video sequence, which refers to the offline association. Methods can either use static or updated graph models [18]. Static graphs refer to traditional approaches that leverage local information among neighboring nodes, while updated graphs use an updating mechanism to capture the global interplay of nodes. The updating mechanism is directly linked to the use of GNNs and their feature aggregation functionality.

Static Graphs NFs are typical among MOT methods [39,16,17] of this category. Object detections and their associations are interpreted with nodes and edges, as shown in Figure 2. In particular, objects are demonstrated with two separated nodes connected by an observation-green edge, which models the observation likelihood, while blue edges, between observations of different frames, express the transition probability. Additionally, the source s and the sink t nodes of NF are connected with all the detections (orange edge), indicating whether a particular detection is the start or end of a trajectory. This is a disjoint paths problem as the desired set of object trajectories (flow paths) should not share any common edges.

The methods that incorporate NF try to identify the set of possible trajectories that can optimally explain the set of input object measurements, which can be expressed theoretically as a Maximum A-Posteriori (MAP) problem [39,17] and practically as a Linear Program (LP) [39,16,17] or an Integer Linear Program (ILP) [23]. The formulation can either be set as a MF [2] or equivalently, a MCF algorithm [39,16,17,23]. Many algorithms can reach a solution, some of which are the push-relabel [39], the simplex [17] or the k-shortest paths [2]. For an approximate solution, a greedy shortest-path algorithm is used by [23] that embeds pre-processing steps, such as Non-Maximum Suppression (NMS), to boost the performance.

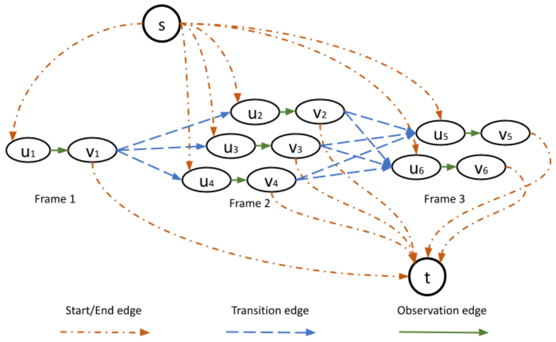


Fig. 2: An NF [39] sample containing six observations spanning over three frames.

Additionally, another method [27] relies on bi-level optimization [6], solving the NF problem and learning the pairwise cost functions for the data association task in an end-to-end manner. Constant cost functions are thus replaced by parameterized ones, allowing the model to be more flexible as it embeds inference into learning. Similarly, the authors in [19] adopt a differentiable NF, but instead of using a local edge loss, they incorporate a global loss during training solving the bi-level optimization problem.

On the other hand, methods [32,33] that treat MOT as an MCMP propose that object detections can be linked not only across time but also space. Unlike NF methods, the possible solutions are now components of the graph instead of paths, which are clustered into different sets that indicate the proposed trajectories. An advantage of these techniques is that no pre or post-processing steps, such as NMS, are required to filter out the detections or the final tracks, allowing it to tackle the challenge of duplicated object detections. Especially in [33], they suggest lifted edges, along with the regular ones, as a way for long-range optimization.

Updated Graphs An attempt to introduce GNNs [26] to the MOT problem is accomplished by [3], where the regular NF formulation becomes fully differentiable. Unlike learning only pairwise costs, this method directly solves the data association problem with edge classification, using a time-aware neural message passing update step [11] to optimize feature representation.

Partially motivated by the probabilistic graph approaches for MOT, [24] proposes an undirected BG, where the association happens dynamically, adding the newest detections and removing the classified or inactive ones. The latter makes the method suitable for online applications, in contrast to other techniques [3], where batches of detections are processed in an offline manner. The difference in the graph construction is that both object detections and their pairwise associations are represented as nodes, which form two different independent and disjoint sets of the BG. Furthermore, an MPNN [11] is performed to update the graph components and circulate information across many timesteps.

Due to the powerful information transmission capability in GNNs, an end-to-end method [21], named Deep Association Network (DAN), introduces a CNN and a Motion Encoder (ME) to extract appearance and motion features, respectively. Subsequently, a GCN [15] associates detections within a batch of frames (local graph optimization) and between consecutive batches of frames (inter-graph optimization).

The MtM category includes methods that most of them perform in an offline manner so as focus on improving the accuracy over the speed. In addition, these methods follow a sliding window approach of variant window sizes that increase the computational complexity as the size of the window broads; however, they contribute to the robustness and better generalization of the results. Further, the use of GNNs in updated graphs improves stability and efficiency.

4.2 Measurement-to-Track Association

The methods that fall into this category endeavor to find a solution for MOT performing BG matching. One popular strategy is to assume a BG and associate new detections with past ones or already created fragments of object trajectories, called tracklets, as shown in Figure 1(b). Since all the methods of this category incorporate GNNs, their graphs are considered updated.

An extension of BG matching is presented in [13], where they update the classical formulation to graph level, performing graph matching in an end-to-end manner. There are two graphs to be matched; the former is constructed by the new detections in a new frame and the second by the existing tracklets. In doing so, they focus on the intra-frame relationships among detections. To this end, they adopt a GCN for feature enhancement and then a differentiable graph matching layer which yields the final matching scores. In another work [36], GNNs are also used to obtain discriminative features that would later benefit both object detection and data association tasks. The association of detections to past tracklets is repeated at every new frame in an online manner.

The paradigm of BG matching is also applied in [22]. Their architecture employs a CNN to extract appearance features from both tracklets and new detections and then a GCN model to update those features through multiple layers. In this case, the association is driven by the sinkhorn algorithm, which is used to normalize the matching scores of the final association matrix. A similar framework is adopted by [14], but instead of using the sinkhorn algorithm as in [22] to satisfy the constraints of one-to-one BG association, they construct a multi-level matrix loss. In [18], two GNNs are designed to elaborate the appearance and the motion features separately. Additionally, each graph network is composed of nodes, edges, and a global variable, with the latter to store past information during tracking. They also introduce four updating mechanisms for the graph components to be updated and to form the connections between detections and already defined objects.

The MtT category meets the real-time response requirement since there has been an improvement in computational complexity as a result of a limited number of frames used in the analysis. Despite the addition of GNNs serves to aggregate past information, improving computational cost alongside performance.

4.3 Track-to-Track Association

This category refers to tracklet association, where different methods [1,23] are firstly used to generate fragments of tracks (tracklets) across a relatively small number of consecutive frames and then stitch tracklets of different time intervals together in a graph. Every step of connecting tracklets together to form longer ones is called association layer and the methods that use this strategy are usually named hierarchical. A variety of formulations is used to solve the data association problem in this category, such as GMMCP, MCMP, or NF.

Performing to a timeframe of a video longer than two consecutive frames (BG match), some approaches [7,25] define a KG to formulate data association, considering all pairwise connections among targets. In the case of [25], the GMCP aims to find the subgraph (clique) that has the least cost compared to the total number of subgraphs in the graph. In the first stage, a video sequence is split into segments, and mid-level tracklets are generated using the GMCP as the best possible clique of a person. Then the resultant tracklets merge again into the final trajectories according to GMCP. An extension of this approach is presented in [7], whereby formulating the association as a Generalized Maximum Multi Clique Problem (GMMCP), they succeed in creating multiple tracks simultaneously instead of finding one at a time [25], following joint optimization. Similarly to [25], the process combines detections of few consecutive frames in low-level tracklets, using overlapping constraints. After the two layers of applying the CMMCP tracker, the final trajectories are ready as a result of merging shorter tracklets to create longer.

In [31], MOT is treated as an MCMP, named Minimum Cost Subgraph Multicut Problem (MCSMP), but instead of using detections as graph components [32,33], they create overlapping tracklet hypotheses according to [1] over a small number of frames. As an alternative to learning the start and end probabilities of an NF, the authors [35] design a simple setup setting an assignment formulation between tracklets. Additionally, they use Siamese CNNs to obtain tracklet affinity features, which are learned concerning a loss function that treats relations of neighboring segments differently than the non-neighboring ones. Given the affinities, the assignment problem is solved using the softassign algorithm [12].

NF in tracklet association uses tracklets as nodes instead of detection responses, enabling long-term object tracking. In [34], they first generate the initial tracklets from the detection set using the NF formulation of [23] and then define an affinity model which estimates the probability (cost) of two tracklets belonging to the same object. The affinity model aims to learn and estimate appearance and motion cues online, i.e. while tracking. Similar to [27], the work in [28] establishes a bi-level optimization, combining feature learning and data association using a learnable NF in an end-to-end framework where tracklets are connected to form the final objects' trajectories.

Table 1: Graph-based methods for Multiple Object Tracking (MOT)

	Method	Year	Graph Opt.	Upt.	Aff. Learn.	Mode	E2E
MtM	MCNF [39]	2008	MCF	x	x	Offline	x
	LP2D [17]	2011	MCF	x	x	Offline	x
	DP_NMS [23]	2011	MCF	x	x	Offline	x
	K_Shortest [2]	2011	MF	x	x	Offline	x
	SiameseCNN [16]	2016	MCF	x	✓	Offline	x
	JMC [32]	2016	MCMP	x	x	Offline	x
	LMP [33]	2017	MCMP	x	✓	Offline	x
	DeepNetFlow [27]	2017	MCF	x	✓	Offline	✓
	DAN [21]	2019	MCF, GCNs	✓	✓	Near-online	✓
	MPNTrack [3]	2020	MCF, GNNs	✓	✓	Offline	x
	TrackMPNN [24]	2021	BG, GNNs	✓	✓	Online	x
	LPT [19]	2022	MCF	x	✓	Offline	x
MtT	EDA_GNN [14]	2019	BG, GNNs	✓	✓	Online	✓
	GCNNMatch [22]	2020	BG, GCNs	✓	✓	Online	✓
	GNMOT [18]	2020	BG, GNNs	✓	✓	Near-online	✓
	GMTracker [13]	2021	BG, GCNs	✓	✓	Online	✓
	GSDT [36]	2021	BG, GNNs	✓	✓	Online	x
TtT	GMCP-Tracker [25]	2012	GMCP	x	x	Offline	x
	GMMCP-Tracker [7]	2015	GMMCP	x	x	Offline	x
	SubgraphMulticut [31]	2015	MCMP	x	x	Offline	x
	CNNTCM [35]	2016	BG	x	✓	Offline	x
	TSML [34]	2016	MCF	x	x	Offline	x
	TAT [28]	2018	MCF	x	✓	Offline	✓

Table legend: Graph Opt. - Graph Optimization; Upt. - Updated graph; Aff. Learn. - Affinity Learning; E2E - End-to-End

In TtT category, methods use tracklets that encode high-level information compared to detections and thus enable long-range tracking, increasing the robustness and the reliability of the results. A particular drawback of this category is that the computational complexity of a method increases alongside the number of association levels.

5 Qualitative Comparison and Discussion

In this section, the classification and the performance of the aforementioned MOT methods are illustrated. Table 1 presents the classification of every method into three categories, namely MtM, MtT, and TtT, sorted by year. The key consideration for this qualitative comparison is to present the similarities/dissimilarities of methods in graph optimization, mode (online/offline), and whether they include updated graphs or affinity learning or are trained end-to-end.

Table 2 depicts the performance evaluation of each MOT method using the four public benchmarks of MOTChallenge [8] 2D MOT2015, MOT16, MOT17, and MOT20. The results were collected from the official MOTChallenge leaderboards or included in the evaluation of the methods in case they are missing from them. The metrics used for the evaluation include MOTA (MOT Accuracy), IDF1 (ID F1-Measure), MT (Mostly Tracked Target Percentage), ML (Mostly Lost Target Percentage), and IDSW (Identity Switch). MOTA is the most important metric since it combines the FP (False Positives), FN (False Negatives), and IDSW. For more details about the metrics see [8].

In general, most of the latest methods belong to the MtT category and adopt BG matching strategies, as well as GNNs, which help update the graph components systematically. By performing online, BG strategies [36,22] are compu-

Table 2: MOT Evaluation on 2D MOT2015/MOT16/MOT17/MOT20 test sets

	Method	MOTA↑	IDF1↑	MT↑ (%)	ML↓ (%)	IDSW↓
MOT15	DP_NMS [23]	14.5	19.7	6.0	40.8	4537
	EDA_GNN [14]	21.8	27.8	9.0	40.2	1488
	SiameseCNN [16]	29.0	34.3	8.5	48.4	639
	CNNTCM [35]	29.6	36.8	11.2	44.0	712
	TSML [34]	34.3	44.1	14.0	39.4	618
	GCNNMatch [22]	46.7	43.2	21.8	28.2	820
	MPNTrack [3]	51.5	58.6	31.2	25.9	**375**
	GSDT [36]*	**60.7**	**64.6**	**47.0**	**10.5**	480
MOT16	DP_NMS [23]	26.2	31.2	4.1	67.5	365
	GMMCP-Tracker [7]	38.1	35.5	8.6	50.9	937
	JMC [32]	46.3	46.3	15.5	39.7	657
	GNMOT [18]*	47.7	43.2	16.1	34.3	1907
	DAN [21]*	48.6	49.3	13.2	43.5	594
	LMP [33]	48.8	51.3	18.2	40.1	481
	TAT [28]	49.0	48.2	19.1	35.7	899
	GCNNMatch [22]	57.2	55.0	22.9	34.0	559
	LPT [19]*	57.4	58.7	22.7	37.2	427
	MPNTrack [3]	58.6	61.7	27.3	34.0	**354**
	GMT_CT [13]	66.2	**70.6**	29.6	30.4	701
	GSDT [36]*	**74.5**	68.1	**41.2**	**17.3**	1229
MOT17	EDA_GNN [14]	45.5	40.5	15.6	40.6	4091
	GNMOT [18]*	50.2	47.0	19.3	32.7	5273
	TAT [28]	51.5	46.9	20.6	35.5	2593
	GCNNMatch [22]	57.3	56.3	24.4	33.4	1911
	LPT [19]*	57.3	57.7	23.3	36.9	1424
	MPNTrack [3]	58.8	61.7	28.8	33.5	**1185**
	GMT_CT [13]	65.0	**68.7**	29.4	31.6	2200
	GSDT [36]*	**73.2**	66.5	**41.7**	**17.5**	3891
MOT20	GCNNMatch [22]	54.5	49.0	32.8	25.5	2038
	MPNTrack [3]	57.6	59.1	38.2	22.5	**1210**
	LPT [19]	57.9	53.5	39.0	22.8	1827
	GSDT [36]*	**67.1**	**67.5**	**53.1**	**13.2**	3133

*MOTChallenge [8] leaderboards do not include these results; they are only available in the evaluation section of the current method.

tationally efficient and suitable for real-time applications, boosting their popularity. MCF formulations [3] have also been devoted to inferring optimal trajectories, offering optimization in a batch of frames or even the entire video, but they are not suitable for real-time applications. In this sense, there is a tradeoff between speed and accuracy, as online strategies provide a faster solution, but since they take into account a small number of frames, they lack robustness and generalization compared to offline methods. This problem can be solved by adding GNNs, as their updated mechanism contributes to information aggregation enabling long-term associations.

Moreover, the latest trends in research have shown a shift from model-based [39,17,23,2] to learning-based methods, as deep learning can benefit both feature extraction and data association. Regarding data association, learning can either be achieved by learning the cost functions [27,28] or by performing edge classification [3]. In the near future, trends are leaning towards end-to-end trainable models [27,22,14,18] as well as architectures that design a joint framework for affinity learning [14] or object detection [36] with graph optimization.

6 Conclusion

This survey condenses a summary and review of graph-based approaches proposed for MOT, in terms of data association, an essential internal part of the tracking-by-detection paradigm that directly impacts a method's performance.

The aim of this review is to organize methods into MtM, MtT and TtT categories, according to how they associate detections and their reflection on the graph structure. Finally, the qualitative and quantitative breakdown of methods in tables offers an insight into the characteristics of each category, providing the reader with all the information needed for future research.

References

1. Andriluka, M., Roth, S., Schiele, B.: People-tracking-by-detection and people-detection-by-tracking. In: CVPR. pp. 1–8. IEEE (2008)
2. Berclaz, J., Fleuret, F., Turetken, E., Fua, P.: Multiple object tracking using k-shortest paths optimization. PAMI **33**(9), 1806–1819 (2011)
3. Brasó, G., Leal-Taixé, L.: Learning a neural solver for multiple object tracking. In: CVPR. pp. 6247–6257. IEEE (2020)
4. Chong, C.Y.: Graph approaches for data association. In: FUSION. pp. 1578–1585. IEEE (2012)
5. Chong, C.Y.: An overview of machine learning methods for multiple target tracking. In: FUSION. pp. 1–9. IEEE (2021)
6. Colson, B., Marcotte, P., Savard, G.: An overview of bilevel optimization. Annals of Operations Research **153**(1), 235–256 (2007)
7. Dehghan, A., Modiri Assari, S., Shah, M.: Gmmcp tracker: Globally optimal generalized maximum multi clique problem for multiple object tracking. In: CVPR. pp. 4091–4099. IEEE (2015)
8. Dendorfer, P., Osep, A., Milan, A., Schindler, K., Cremers, D., Reid, I., Roth, S., Leal-Taixé, L.: Motchallenge: A benchmark for single-camera multiple target tracking. IJCV **129**(4), 845–881 (2021)
9. Emami, P., Pardalos, P.M., Elefteriadou, L., Ranka, S.: Machine learning methods for data association in multi-object tracking. CSUR **53**(4), 1–34 (2020)
10. Fan, L., Wang, Z., Cail, B., Tao, C., Zhang, Z., Wang, Y., Li, S., Huang, F., Fu, S., Zhang, F.: A survey on multiple object tracking algorithm. In: ICIA. pp. 1855–1862. IEEE (2016)
11. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Neural message passing for quantum chemistry. In: ICML. pp. 1263–1272. PMLR (2017)
12. Gold, S., Rangarajan, A., et al.: Softmax to softassign: Neural network algorithms for combinatorial optimization. Artificial Neural Networks pp. 381–399 (1996)
13. He, J., Huang, Z., Wang, N., Zhang, Z.: Learnable graph matching: Incorporating graph partitioning with deep feature learning for multiple object tracking. In: CVPR. pp. 5299–5309. IEEE (2021)
14. Jiang, X., Li, P., Li, Y., Zhen, X.: Graph neural based end-to-end data association framework for online multiple-object tracking. arXiv:1907.05315 (2019)
15. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv:1609.02907 (2016)
16. Leal-Taixé, L., Canton-Ferrer, C., Schindler, K.: Learning by tracking: Siamese cnn for robust target association. In: CVPR workshops. pp. 33–40. IEEE (2016)
17. Leal-Taixé, L., Pons-Moll, G., Rosenhahn, B.: Everybody needs somebody: Modeling social and grouping behavior on a linear programming multiple people tracker. In: ICCV workshops. pp. 120–127. IEEE (2011)
18. Li, J., Gao, X., Jiang, T.: Graph networks for multiple object tracking. In: WACV. pp. 719–728. IEEE (2020)

19. Li, S., Kong, Y., Rezatofighi, H.: Learning of global objective for network flow in multi-object tracking. In: CVPR. pp. 8855–8865 (2022)
20. Luo, W., Xing, J., Milan, A., Zhang, X., Liu, W., Kim, T.K.: Multiple object tracking: A literature review. Artificial Intelligence **293**, 103448 (2021)
21. Ma, C., Li, Y., Yang, F., Zhang, Z., Zhuang, Y., Jia, H., Xie, X.: Deep association: End-to-end graph-based learning for multiple object tracking with conv-graph neural network. In: ICMR. pp. 253–261. ACM (2019)
22. Papakis, I., Sarkar, A., Karpatne, A.: Gcnnmatch: Graph convolutional neural networks for multi-object tracking via sinkhorn normalization. arXiv:2010.00067 (2020)
23. Pirsiavash, H., Ramanan, D., Fowlkes, C.C.: Globally-optimal greedy algorithms for tracking a variable number of objects. In: CVPR. pp. 1201–1208. IEEE (2011)
24. Rangesh, A., Maheshwari, P., Gebre, M., Mhatre, S., Ramezani, V., Trivedi, M.M.: Trackmpnn: A message passing graph neural architecture for multi-object tracking. arXiv:2101.04206 (2021)
25. Roshan Zamir, A., Dehghan, A., Shah, M.: Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs. In: ECCV. pp. 343–356. Springer (2012)
26. Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M., Monfardini, G.: The graph neural network model. Neural Networks **20**(1), 61–80 (2008)
27. Schulter, S., Vernaza, P., Choi, W., Chandraker, M.: Deep network flow for multi-object tracking. In: CVPR. pp. 6951–6960. IEEE (2017)
28. Shen, H., Huang, L., Huang, C., Xu, W.: Tracklet association tracker: An end-to-end learning-based association approach for multi-object tracking. arXiv:1808.01562 (2018)
29. Singh, T., Vishwakarma, D.K.: Human activity recognition in video benchmarks: A survey. Advances in Signal Processing and Communication pp. 247–259 (2019)
30. Sun, Z., Chen, J., Chao, L., Ruan, W., Mukherjee, M.: A survey of multiple pedestrian tracking based on tracking-by-detection framework. CSVT **31**(5), 1819–1833 (2020)
31. Tang, S., Andres, B., Andriluka, M., Schiele, B.: Subgraph decomposition for multi-target tracking. In: CVPR. pp. 5033–5041. IEEE (2015)
32. Tang, S., Andres, B., Andriluka, M., Schiele, B.: Multi-person tracking by multicut and deep matching. In: ECCV. pp. 100–111. Springer (2016)
33. Tang, S., Andriluka, M., Andres, B., Schiele, B.: Multiple people tracking by lifted multicut and person re-identification. In: CVPR. pp. 3539–3548. IEEE (2017)
34. Wang, B., Wang, G., Chan, K.L., Wang, L.: Tracklet association by online target-specific metric learning and coherent dynamics estimation. PAMI **39**(3), 589–602 (2016)
35. Wang, B., Wang, L., Shuai, B., Zuo, Z., Liu, T., Chan, K.L., Wang, G.: Joint learning of siamese cnns and temporally constrained metrics for tracklet association. arXiv:1605.04502 (2016)
36. Wang, Y., Kitani, K., Weng, X.: Joint object detection and multi-object tracking with graph neural networks. In: ICRA. pp. 13708–13715. IEEE (2021)
37. Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Philip, S.Y.: A comprehensive survey on graph neural networks. Neural Networks and Learning Systems **32**(1), 4–24 (2020)
38. Xu, Y., Zhou, X., Chen, S., Li, F.: Deep learning for multiple object tracking: a survey. IET Computer Vision **13**(4), 355–368 (2019)
39. Zhang, L., Li, Y., Nevatia, R.: Global data association for multi-object tracking using network flows. In: CVPR. pp. 1–8. IEEE (2008)