This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3049397, IEEE Transactions on Circuits and Systems for Video Technology

A Graphical Social Topology Model for RGB-D Multi-Person Tracking

Shan Gao, *Member, IEEE*, Qixiang Ye, *Senior Member, IEEE*, Li Liu, *Member, IEEE*, Arjan Kuijper, *Member, IEEE*, Xiangyang Ji, *Member, IEEE*

***Abstract*—Tracking multiple persons is a challenging task especially when persons move in groups and occlude one another. Existing research have investigated the problems of group division and segmentation; however, lacking overall person-group topology modeling limits the ability to handle complex person and group dynamics. We propose a Graphical Social Topology (GST) model in the RGB-D data domain, and estimate object group dynamics by jointly modeling the group structure and states of persons using RGB-D topological representation. With our topology representation, moving persons are not only assigned to groups, but also dynamically connected with each other, which enables in-group individuals to be correctively associated and the cohesion of each group to be precisely modeled. Using the learned typical topology pattern and group online update modules, we infer the birth/death and merging/splitting of dynamic groups. With the GST model, the proposed multi-person tracker can naturally facilitate the occlusion problem by treating the occluded object and other in-group members as a whole, while leveraging overall state transition. Experiments on different RGB-D and RGB datasets confirm that the proposed multi-person tracker improves the state-of-the-arts.**

***Index Terms*—RGB-D Multi-Person tracking, topology model, group behavior analysis**

I. Introduction

Multi-object Tracking (MOT) is a fundamental problem in computer vision and contributes to many applications, including video surveillance [1]–[5], intelligent vehicles [6], [7], and robotics [8], [9]. MOT has received increasing attention and many approaches [10], [11], have been proposed to tackle this problem with considerable progress. However, this problem is far from being solved due to factors such as complex dynamics, abrupt appearance changes, and severe object occlusions, especially when the target objects are moving persons. In this paper, we mainly focus on Multiple Person Tracking (MPT), which is a task of predicting trajectories of all person instances in a video. Conventional RGB-based MPT methods that optimally link person detections with respect to their appearance, motion, and time gap, have been intensively investigated [10], [11]; however, modeling complex dynamics and target occlusions are often beyond the scope of their confines and capabilities.

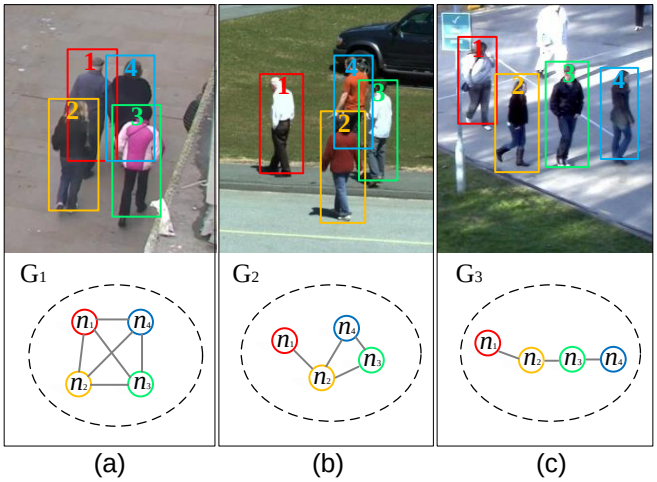


Fig. 1. The motivation of using a GST model. The groups G_1, G_2, and G_3 are identical if considered as a node set of indices, where each node represents one person. But from the view of topology configuration, G_1 contains more edges than G_2 and G_3, which means a stronger connection and a similar movement among persons. In tracking process, G_2 and G_3 structures are more likely to split than G_1. The GST model with topology changes thus allows propagating more information than a set of indices. In this paper, we aim to investigate the topology changes in and out groups, and bridge the gap between the group and individual tracking.

To address the occlusion problem, depth information obtained from stereo cameras has been widely used in MPT [6], [12]. In [6], [12], Ess *et al*. proposed a joint estimation approach based on a tracking-by-detection framework for multi-person tracking in busy environments from a synchronized camera pair. Depth estimation provided by the stereo pair allows stereo-based methods to achieve good results in challenging scenarios, but if considering time of depth estimation and person detection, the stereo-based MPT algorithms are not efficient. The long time consumption makes stereo cameras hard, if not impossible, to become an efficient end-to-end detection and tracking system.

There are two kinds of acquisition sets which provide the depth data directly. The first one is the RGB-D sensor which utilizes reliable and affordable RGB-D sensors, such as Microsoft Kinect and Intel RealSense, thereby enabling reliable RGB-D data acquisition with low cost, high efficiency, and high quality, thus further advancing RGB-D related research in computer vision [13], [14]. The second one is LIDAR

Shan Gao is with the Unmanned System Research Institute at Northwestern Polytechnical University, Xi'an, and he is also with Tsinghua University, China. (Email: gaoshan@nwpu.edu.cn)

Qixiang Ye is with the School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, China. (Email: qxye@ucas.ac.cn)

Li Liu is with College of System Engineering, National University of Defense Technology, China and she is also with University of Oulu, Finland.

Arjan Kuijper is with Fraunhofer Institute for Computer Graphics Research (IGD) and Technology University of Darmstadt, Germany. (Email: arjan.kuijper@mavc.tu-darmstadt.de)

Xiangyang Ji is with the Department of Automation at Tsinghua University, Beijing, China. (Email: xyji@tsinghua.edu.cn)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3049397, IEEE Transactions on Circuits and Systems for Video Technology

(*e.g.*, Velodyne), which provides an accurate dense 3D point cloud, replacing the Kinect in outdoor driving platforms. New registration methods [7], [15] combine dense point clouds and RGB images, interpolate a sparse set of pixels, and provide a dense map where each pixel has an associated depth value.

With RGB-D sensors, the occlusion problem in sparsely populated scenes has been rectified. In crowded scenes where objects have complex dynamics; however, the MPT problem remains unsolved. To model persons' dynamics in a crowd, social behavior analysis [16]–[18] has recently been explored. Sociologists observe that about 70% of the persons walk in the form of the group. Persons within the same group has a similar motion and to be close to one another, perhaps subconsciously encouraging group interaction. The group-level MPT methods [19]–[23], as opposed to conventional MPT, aim to detect and track groups of persons that share spatial-temporal characteristics (*i.e.*, velocity, range, and geographical goal). In crowded scenes, however, the number and structure of groups vary over time as persons might enter or leave a scene, randomly. Groups can also split, merge, be relatively close to one another or move independently through a crowd, which highlights the complexity involved in developing this analytical framework. The group-level MPT methods are reasonably competent for characterizing, detecting, and tracking groups, although, few [20]–[23] comprehensively model group dynamics from the dynamic group structures or evolving topographical perspective.

Towards constructing a general multi-person tracker applicable for both RGB and depth data, we propose a novel GST model to quantify group dynamics in an RGB-D domain, aiming to track in- and out-group movement accurately. We statistically infer which persons move in formation or have common movement, as well as modelling behaviors, within and between groups. This information accompanies MPT applications where the goal is to differentiate in-group members from out-group persons, or to predict the intention, destination, and future manoeuvres of objects. The motivation for GST (*cf*. Fig. 1) lays with the possibility of using common group information to improve the tracking of individual persons as well as using topology configurations to infer the birth/death and merging/splitting of dynamic groups.

To sum up, the main contributions of this study to the field are, as follows:

- A GST model. The social affinity in natural crowds is quantified according to topological modelling. This topology relations analysis is formulated using strong contextual information to infer group and individual states.
- An RGB-D based group learning strategy. This strategy integrates birth, update, merge, and split modules to topologize group dynamics. Aggregated with the trained typical topology patterns, this learning strategy facilitates the description of topology transformation.
- A unified group and individual joint tracking framework. This fills the gap between group modeling and MPT by simultaneously identifying the group and individuals within.

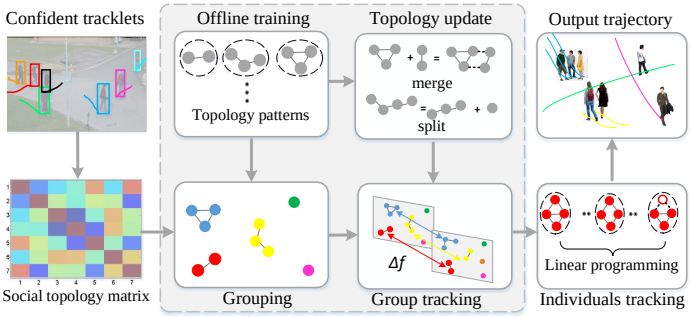


Fig. 2. The framework of the proposed graphical social topology model. First, we group the clusters based on confident tracklets using the social affinity matrix. The grouping is then aggregated with the topology management with the movement of persons, in which a joint group and individual tracking method is proposed to solve the data association problem. Finally, we use linear programming to solve the individual tracking and output the complete trajectories.

By integrating the proposed GST model with the classical data association method, we develop an MPT algorithm can be applied directly to both RGB and RGB-D data.

II. Related Work

In this section, we review the most relevant tasks of the group and individual tracking in computer vision, including multiple person tracking, RGB-D tracking, group modeling, and group tracking.

Recently, CNN-based MOT methods [24]–[28] have attracted increasing attentions. The CNNs architecture has been used for modeling appearance. The high-level feature is extracted by CNNs trained for a specific task. Sadeghian *et al*. [25] designed a recurrent neural network combining the deep features to track the unreliable detections; Chu *et al*. [24] used the attention mechanism to track the long trajectory; Hilke *et al*. [26] performed the detection and tracking in a single neural network architecture instead of heuristic decisions over the track lifetime; Bae *et al*. [27] combined online transfer learning to improve appearance discriminability by adapting the pre-trained deep model during online tracking. In general, deep networks can be designed as on-line appearance classifiers to discriminate targets from backgrounds. The deep appearance learning methods learn discriminative appearance models from large training datasets, since the conventional appearance learning methods do not provide a rich representation that can distinguish multiple persons with large appearance variations. However, in the RGB-D domain, the RGB-D appearance feature is not in the scope of the above CNN-based methods. In this paper, our graphical social topology model is based on the social context information among the objects, exploring the social forced impact in RGB-D multiple person tracking.

RGB-D tracking aims at achieving real-time tracking integrated with the RGB-D detection. In [29], an SVM classifier is trained using HOG features extracted in color and depth frames, large displacement optical flow is integrated, and occlusion is handled by assuming that the target is the closest object in the bounding box when there is no occlusion. In [30], Munaro *et al*. used a depth-based sub-clustering method for tracking people within groups or near the background and a

joint likelihood for decrease drifts and ID switches problem. In [7], [31], Gao *et al*. proposed real-time graph models to infer multiple objects according to the RGB, motion, and depth domains. In [8], Linder *et al*. proposed a fully integrated real-time multi-modal RGB-D people tracking system for moving platforms in crowded environments. These works present real-time single and/or multi-object tracking systems. Nevertheless, few of them considers the interaction between objects, and therefore lack a strong social context of moving objects.

Social context has been studied intensively in this decade. Researchers target to find a stable and accurate clustering way to describe movement in the form of groups. These studies provide the trajectory-level analysis to model and discovery groups in crowded and semi-crowed scenes. These group modeling methods, together with social behavior research, formulated as social force models [16], [32], are used as high-level constraints and have attracted increasing attention in the MPT framework. Pellegrini *et al*. [19] proposed an effective dynamic group model, considering nearby pedestrians' positions. Qin and Shelton [33] used a dual optimization framework and a linear programming solution to model the social group behavior as a high-level clue. Bazzani *et al*. [34] assumed a tight relation of mutual support between the modeling of individuals and groups, promoting the idea that groups are better modeled if individuals are considered, and vice versa. These group-based tracking methods confirm that the group model in MPT framework could be a strong and effective constraint. The proposed GST model uses a topology graph to represent the group update, merge, and split flexibly, nevertheless, it is not pure energy or cost optimization manner solved in a local search manner.

In order to create a feasible comparison to [23], [33], [35], we propose to utilize social grouping information as natural and flexible high-level constraints (cf. Fig. 2). Our study therefore, mainly focuses on dashed rectangles and how to design and use effective topolographical information to garner insight into group and individual dynamics from a social forces perspective. Accordingly, our work focuses on using analytical social groups to maintain individual identities by the group and individual tracking in the RGB-D domain.

III. Overview

The proposed method falls into the tracking-by-detection framework. Suppose, a set of tracklets $\mathcal{L} = (n_1, \cdots, n_n)$ is generated through a video sequence and each tracklet (n_i) is either a consecutive sequence of detection responses or interpolated responses which contains the same person. Given certain spatial-temporal constraints, the goal of the MPT is to determine which tracklets correspond to that person. The RGB tracking methods calculate tracklet affinity, solely within the RGB domain. Our MPT model is based on these RGB-D tracklets.

First, we propose an RGB-D tracklet generation method to associate the detection results (the detection results come from an RGB-D detector, also, could be provided within datasets). In each frame f, the tracklet n_i^f, corresponding to the person i, is represented by a set of state variables $n_i^f = (a_i^f, X_i^f, v_i^f, o_i^f)$, where a_i^f, X_i^f, v_i^f, and o_i^f denotes the appearance, position, speed, and orientation, respectively. We calculate the tracklets social affinity, as a social topology matrix based on the tracklets, to estimate group dynamics and spatial topology, which supports learning typical topographical group patterns in the sequence found within groups of people. These learned group patterns are regarded as the reference group during the initialization of group tracking. Further, we design the birth, death, merging, and splitting modules to reflect group evolution processes during tracking. Finally, joint group-individual tracking is proposed to identify the same groups in the sequence and associate individuals within groups. The proposed model is provided in Fig. 2, with pertinent notations for the model, described in Table I.

TABLE I
NOTATIONS OF THE VARIABLES

Symbol	Description
T	social affinity matrix $[T_{ij}]$
n_i	node/tracklet, $n_i = (a_i, X_i, v_i, o_i)$
l_i	motion vector of n_i, $l_i = (X_i, v_i)$
X_i	location vector of n_i, $X_i = (x_i, y_i, z_i)$
G	group $\{G_k\}$
E	edge set in G
N_k	the size of group G_k: number of nodes in G_k
g_k	the center of group G_k
π_i	topological representation of n_i in group, $\pi_i = (r_i, \theta_i)$
$D_{n_i}^k$	degree of node n_i in G_k
C	sampling matrix in individuals
B	sampling matrix in group
L_{ij}	co-existing period between n_i and n_j
A_{ij}	appearance affinity between n_i and n_j
Ψ_{ij}	location affinity between n_i and n_j
M	binary association matrix, $M = [M_{ij}]$
α	weighting factor (Eq. 3)
λ	distance threshold (Eq. 4)
l	co-existing period threshold (Eq. 5)
τ	edge weight threshold (Sec. IV-E)

IV. RGB-D Graphical Social Topology Model

A. RGB-D Tracklet Generation

We extract the RGB-D feature for the object appearance. Given a video with depth data, the combined features in the Region Of Interest (ROI) is extracted to describe the object's appearance and 3D position. Assuming that each object is an isolated 3D bounding box, a set of RGB-D based features is extracted, including Histograms of Oriented Gradient and Color (HOGC) features [36], and Histogram of Oriented Depth (HOD) features [9], [37]) to discriminate objects from the background.

RGB-D data project onto the X-Y and the Y-Z planes. The Y-Z plane is an auxiliary plane, where the average depth value is calculated for one object. We define this object and the corresponding background seeds as a set of pixels inside the ROI. To obtain object seeds in the projected 2D bounding box, we remove these pixels corresponding to the background seeds. In addition, we utilize an online adaptive feature pool, the 14*7 HOGC [23] feature. In the X-Z plane, we extract 9*5 variation feature bins [37] on the point clouds locations. There are totally 143 bins of RGB-D features a_i in total to represent

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3049397, IEEE Transactions on Circuits and Systems for Video Technology

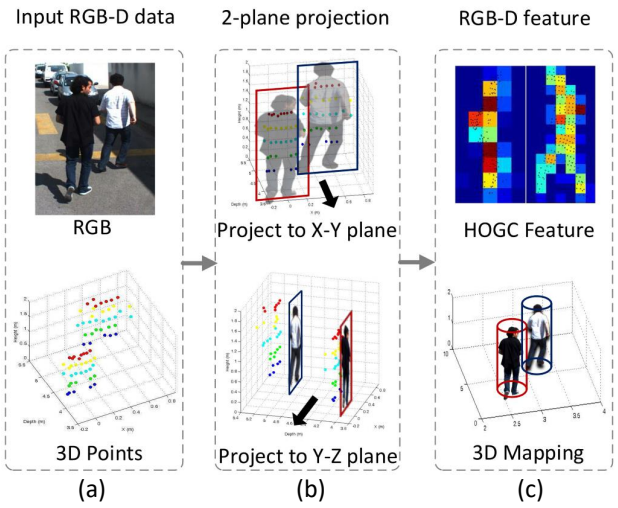


Fig. 3. RGB-D feature extraction [23]. Best viewed in color.

an objects n_i. Then the appearance affinity A_{ij} between n_i and n_j is defined as

$$A_{ij} = exp(-1/c^2 \cdot L^2(n_i, n_j)^2), \tag{1}$$

$$L^2(n_i, n_j) = \|a_i - a_j\|_2^2, \tag{2}$$

where L^2 computes the square of the $L2$ norm, and c is a normalization factor.

In the initialization of the RGB-D tracklets generation, a set of tracklets is generated after a low-level association in an overlapping manner. Targets on each frame are detected using a pre-trained detector [30], [38]. Similar to [7], [31], a nearest neighbor detection association method is adopted to generate the initial tracklets. For each unassociated detection, a Kalman filter based tracker is initialized with position and velocity states. A detection A is associated with a detection B in the next frame if B has the minimum distance to the predicted location and overlaps at least 50% (measured as $size(A \cap B)/size(A \cup B)$ in size with detection A. The corresponding Kalman filter is then updated with the newly associated detection. The tracklets generation terminates if no association is found for more than two consecutive frames, or one detection is associated by multiple tracklets.

B. Social Topology Matrix

Based on the RGB-D tracklets, we define a social affinity matrix $T = [T_{ij}]$ to measure the social affinity between tracklets n_i and n_j. Note that only confident tracklets are considered for grouping analysis, as there might be false alarms and incorrect associations in the input tracklets. We define a tracklet as a confident one if it is long enough (more than l frames) since most false tracklets are short. The social affinity T_{ij} between two confident tracklets is then given by

$$T = \alpha_d T_d + \alpha_t T_t + \alpha_v T_v + \alpha_o T_o, \tag{3}$$

where T_d, T_t, T_v, and T_o are the social affinities based on distance, time, speed, and orientation at frame f[1]. α_d, α_t, α_v, and α_o are weighting factors.

[1]The superscript f is omitted for simplicity.

Distance. Social behavior analysis shows that pedestrians tend to unconsciously organize the space around them in particular configurations with different degrees of intimacy [18]. The shorter the distance between two persons, the higher the degree of intimacy. We adopt different distance measure strategies according to the states of objects in datasets. In RGB-D datasets with the real-world depth information, d_{ij} denotes the world-coordinate average distance between two persons. In RGB datasets, d_{ij} denotes the pixel distance between the center points of the persons' bounding boxes in the images. The distance affinity is defined as

$$T_d(n_i, n_j) = \frac{\lambda}{2d_{ij}}. \tag{4}$$

Here, we define a distance threshold $\lambda = w_i + w_j$, where w_i and w_j are the width of the person i, and j, respectively. λ is learned on training datasets, which will be detailed in Sec. IV-F.

Time. It is observed that members in the same group usually appear and disappear at a similar time. The time term indicates how long two tracklets n_i and n_j perform similar movement and stay close to each other. Let L_{ij} denote the length of the co-existing period. The time affinity between a pairwise of tracklets is defined as

$$T_t(n_i, n_j) = \frac{2L_{ij}}{L_{ij} + l}, \tag{5}$$

where l is the co-existing period threshold of two tracklets. This threshold guarantees that two tracklets n_i and n_j last for at least l frames.

Speed. Persons in a group tend to have the same speed. Let v_i and v_j denote the speeds of tracklets n_i and n_j respectively. The speed affinity between n_i and n_j is defined as

$$T_v(n_i, n_j) = \mathcal{N}(\|v_i - v_j\|), \tag{6}$$

where $\mathcal{N}(\cdot)$ is a min-max normalization operator applied independently for each pairwise tracklets to linearly scale their speed differences into the range $[0, 1]$.

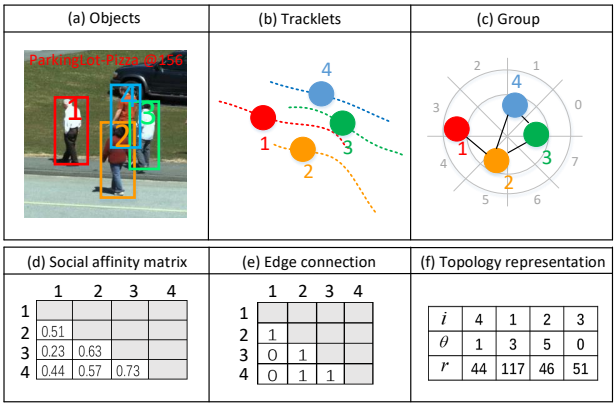
Orientation. We adopt an improved Potts model similar to [39] to define the affinity among different moving orientations as

$$T_o(n_i, n_j) = \frac{1 + \cos(o_i - o_j)}{2}, \tag{7}$$

where $o_b = \frac{2\pi q_b}{q}$ and $b = i, j$. The person's moving orientations are quantified into q bins. Here we use $q = 8$, which means a resolution of $45°$ from '0' to '7' between neighboring orientation bins. Additional '8' means that the person stands still. It is the neighbor bin of any orientation bin. q_b is the moving orientation.

We calculate the speed and orientation factors in two terms. This enables the effectiveness of the social affinity against poor detections. Especially, the '8' orientation bin is assigned to a stationary object, which causes the stationary pairwise tracklets to keep a stable social affinity. The social topology matrix can also be employed as a tool to describe the group dynamics for different applications. We apply the social topology matrix to group and individual tracking in this paper.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3049397, IEEE Transactions on Circuits and Systems for Video Technology



	1	2	3	4
1				
2	0.51			
3	0.23	0.63		
4	0.44	0.57	0.73	

	1	2	3	4
1				
2	1			
3	0	1		
4	0	1	1	

i	4	1	2	3
θ	1	3	5	0
r	44	117	46	51

Fig. 4. Representation of objects in different measurements. (a) objects in bounding boxes; (b) objects in tracklets; (c) objects in a topological group; (d) social affinity matrix among objects in a group; (e) edge connection among objects in a group; (f) topological description of objects.

C. Topological Representation

Given a social affinity matrix T, a graph is defined as $G = (\{n_1, \cdots, n_N\}, E(T_{ij}))$, wherein N objects constitute the set of nodes $\{n_1, \cdots, n_N\}$ linked by edge set E, and each node n_i is associated with the tracklet of one object (one tracklet equals one node in graph). For instance, using this definition, the Groups G_1, G_2 and G_3 in Fig. 1 can be denoted as $G_1 = (\{n_1, n_2, n_3, n_4\}, \{T_{12}, T_{13}, T_{14}, T_{23}, T_{24}, T_{34}\})$, $G_2 = (\{n_1, n_2, n_3, n_4\}, \{T_{12}, T_{23}, T_{24}, T_{34}\})$ and $G_3 = (\{n_1, n_2, n_3, n_4\}, \{T_{12}, T_{23}, T_{34}\})$, respectively. As the weight of the edge set E is measured by the dynamic social topology affinity matrix T (Eq. 3), G is a dynamic topological graph representing the structure of the in-group members. Fig. 4 visualizes objects in different measurements; (a)-(c) show the objects in the image, as tracklets, and in the group. The edge connection relation in (c) is decided by the social affinity matrix in (d). If T_{ij} is larger than the threshold τ, there is an edge between them and the according element in (e) is 1, and vice versa.

In order to describe the group in a whole moving unity, we record a virtual group center $g_k = \frac{1}{N_k}\sum X_i$, where $X_i \in G_k$. The center and the covariance matrix of each group can be characterized differently, *e.g.*, based on a mixture of Gaussian components. We represent the topology of a group as $\pi_i = (r_i, \theta_i)$, where r_i represents the distance between a node and the virtual center. $\theta_i = tan^{-1}\frac{|z_i - z_k|}{|x_i - x_k|}$ denotes the topology orientation bin within a group, which is quantified in the interval [0,7] and $\theta_i \in \mathbb{Z}$ (see in Fig. 4(f)). With the affinity matrix T and the topology relation π, G is flexible to update the group structure and model the interaction among group members.

D. Social Topology Property

We introduce two properties, *i.e.*, *compactness* and *consistency*, of a social topology. The compactness property quantifies the spatial structure of the topology. The consistency property describes the temporal and spatial evolvement of the in-topology members. Such properties enable the social topology to handle group management flexibly, including splitting and merging.

Compactness. In graph theory, the degree of a node in a graph is the number of connections (or edges) it has to other nodes. Here we adopt the 'degree' to define the group compactness to measure the edge density in a group. Let D_{n_i} record the number of edges incident to n_i, and D^k the total degree of all the nodes in G_k. The compactness constraints are defined as

$$\begin{cases} I : D^k > 2(N_k - 1), \\ II : \max D_{n_i} = N_k - 1, \end{cases} \tag{8}$$

where N_k is the size of a graph recording the number of the nodes in the graph. Constraint *I* guarantees that the topology has a high edge density. This can exclude the groups with a 'line-like' topology, such as G_3 in Fig. 1(c). Constraint *II* guarantees a compact structure. This enables the in-group members to distribute around a center member, *i.e.*, a 'star-like' topology. A qualified topology configuration of a group should satisfy at least one of the above two compactness constraints. (Fig. 5 visualizes the typical group topologies satisfying constraints *I* and/or *II*.)

Consistency. We define topology consistency to represent in-group spatial evolvement in sequences. When tracking target groups, each node n_i is characterized by the corresponding motion vector $l_i = (x_i; \tilde{x}_i; y_i; \tilde{y}_i; z_i; \tilde{z}_i)^T$, where $X_i = (x_i; y_i; z_i)$ represents the vector position and $v_i = (\tilde{x}_i; \tilde{y}_i; \tilde{z}_i)$ the vector velocity. Here, each target is considered a specific point and its motion is assumed to be positioned along the horizontal X-Z plane. So, the motion vector is calculated as $l_i = (x_i; \tilde{x}_i; z_i; \tilde{z}_i)^T$, and the state of the ith target is given by:

$$l_i^f = C l_i^{f-F} + \Gamma \varphi^{f-F}, \tag{9}$$

where

$$C = diag(C_1, C_1),$$
$$C_1 = \begin{pmatrix} 1 & F \\ 0 & 1 \end{pmatrix},$$
$$\Gamma = \begin{pmatrix} F/2 & 1 & 0 & 0 \\ 0 & 1 & F/2 & 1 \end{pmatrix}^T.$$

Here, F is the time interval between two groups, and φ is the system dynamics noise. Considering the nonlinear motion, especially the abrupt speed and orientation variation in the tracking, the system dynamics noise φ is represented as a sum of two Gaussian components $p(\varphi) = \eta\mathcal{G}(0, Q_1) + (1-\eta)\mathcal{G}(0, Q_2)$, where $Q_1 = diag(\sigma^2, \sigma_1^2)$ and $Q_2 = diag(\sigma^2, \sigma_2^2)$. σ is a standard deviation assumed to be constant for x and y. $\sigma_1 \ll \sigma_2$ are standard deviations allowing smooth and abrupt changes in the velocity, respectively. The coefficient η has values in the interval $[0, 1]$ (we set $\eta = 0.7$ in the experiments). C and Γ control the motion state of a target, where C defines the motion update interval and Γ controls the noise update. More sophisticated models can be adopted to model the targets interactions in each group, such as the developed in [40].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3049397, IEEE Transactions on Circuits and Systems for Video Technology

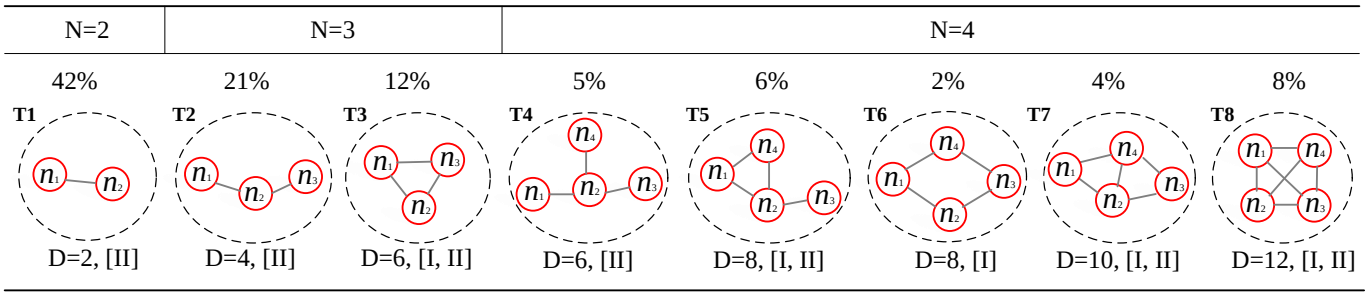


Fig. 5. The learned basic topology patterns. N and D record the size and total degree of a topology pattern, $[\cdot]$ denotes which compactness constraint the topology pattern satisfies in Eq. 8. The frequency of each topology pattern in the training datasets is shown above each pattern.

E. Group Update

We formulate the topology online update process by four submodules: Birth, State Update, Merge, and Split, as described in Algorithm 1. Compared with other clustering and inference methods, the proposed model is able to automatically discover the number of groups and perform updating, merging, and splitting with the dynamic social topology.

Birth. Initially, the edge set is empty, $E = \{\emptyset\}$, and the social affinity T_{ij} is used as the edge weight. There is an edge between n_i and n_j if $T_{ij} > \tau$, where τ is the edge threshold, the learning of which is introduced in Sec. IV-F. We then group the connected nodes according to eight typical topology patterns, as shown in Fig. 5 (*cf.* Sec. IV-F). If the topology of the connected nodes matches one of the typical topology patterns, we merge them as one group. In the Birth-Module, we do not initialize groups with a large size. This avoids the large-size topology with a false tight structure because large-size groups might easily split up in the following frames. Conversely, small groups are assembled in the Merge-Module when they perform a high social affinity.

State Update. The topology structure of a group is related to the social affinity in tracking, so the existing edges should be updated by the social affinity T_{ij}. Further, when T_{ij} between two nodes from different groups is more than the edge threshold τ, the management goes to the Merge-Module. When T_{ij} between in-group members is less than the edge threshold τ, the management goes to the Split-Module. Each edge by T_{ij} between the pairwise nodes is updated by the state transition matrix C in this topology. Moreover, the Update-Module is utilized to solve the *self-occlusion* problem inside a group (details in Sec. V-B).

Merge. When two groups move close to each other and the states last for l frames, they are merged into a big group. It should be noticed that not all the groups moving close ($T_{ij} > \tau$) can be merged together. The reasons are that the topology of merged group is expected to be as compact as possible, and the new group should satisfy the compactness constraint defined in Eq. 8. The total degree of the merged group is $D = D_{G1} + D_{G2} + D_{new}$, where D_{new} is the added degree generated by new edges between two groups. We define the merging constraint as

$$D_{new} > \min(N_1, N_2), \tag{10}$$

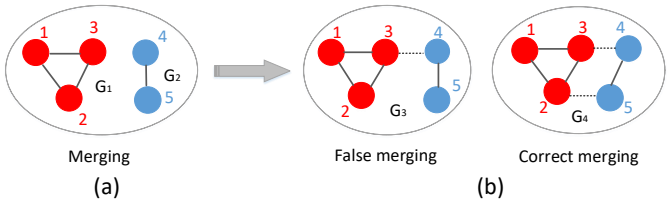


Fig. 6. An example for the false and correct merging case. G_3 and G_4 are new groups consisting of G_1 and G_2. The dash lines are new edges. Only G_4 satisfies the merging constraint: the D_{new} of G_4 is larger than $\min(N_1, N_2) = 2$, according to Eq. 10.

where N_1 and N_2 are the size of G_1 and G_2, respectively. D_{new} should be larger than the size of the smaller group, *i.e.* $\min(N_1, N_2)$. An example is shown in Fig. 6.

Split. When the members inside a group move at different velocities and/or orientations, this group is considered to split into small groups. Under this circumstance, the edge weights between members increase largely, and the edge is removed when T_{ij} is lower than the threshold τ.

The above group management scheme has the following advantages. First, one can redesign and debug individual components effectively. Second, the modular framework makes it easy to replace each module or to insert a new one. Third, it enables separate training of each module, speeding up the model training.

F. Topology Pattern Learning

We investigate spatial organization of walking pedestrian topology to determine whether there are typical patterns within spatial topological configuration. Such topology patterns are used in groups initialization and update.

Given a set of detections and the corresponding Ground Truth (GT) in the training sequence, the GT IDs are first assigned to each person as complete trajectories. Assuming there are N people (*i.e.*, trajectories) in one frame and the co-existing times are longer than l frames. We therefore calculate the $N \times N$ social topology matrix T according to Eqs. 3-7 and achieve a fully connected graph, where edge weights are the social topology affinity T_{ij}. In clips of the sequences (which have groups of people within), we use a semi-unsupervised method to identify the basic group pattern and the optimal parameter settings. We give an augmented group a numerical value $N \in \{2, 3, 4\}$ in order to identify a typical group

Algorithm 1: Group update

• Birth
Input: $E = \{\emptyset\}$
Output: $G = \{G_k\}$
Step-1:
  Calculate $T = [T_{ij}]$ in Eq. 3;
  For each connection T_{ij};
    IF $T_{ij} > \tau$, $E = E \cup \{(n_i, n_j)\}$;
  End for
Step-2:
  Cluster E and generate the coarse group set $\{G_k\}$;
  For each group G_k
    IF D_k dissatisfies Eq. 8, **GOTO** Split-Module;
    Regulate the topology by typical topology patterns in Fig. 5;
  End for

• State Update
Input: $G^{f-1} = \left\{G_k^{f-1}\right\}$, $T^f = [T_{ij}^f]$
Output: $G^f = \left\{{G_k}^f\right\}$
For each group G_k^{f-1}
Step-1:
  Update edge weights by T_{ij}, $i, j \in G_k$;
Step-2:
  IF group size $N_k^f < N_k^{f-1}$, ADD virtual nodes (Sec. V-C);
  IF edges connect with other groups $T_{ij} > \tau$, **GOTO** Merge-Module;
  IF the compactness D_k dissatisfies Eq. 8, **GOTO** Split-Module;
End for

• Merge
Input: G_{k1}, G_{k2}
  Calculate the compactness D_{new} of group $G_{k1} \cup G_{k2}$;
  If $D_{new} > \min(N_{k1}, N_{k2})$
    Output: $G_{new} = G_{k1} \cup G_{k2}$;
  Else
    Output: G_{k1} and G_{k2};

• Split
Input: G_k, T
Output: $\left\{G_k^{'}\right\}$
  While D_k dissatisfies compactness constraints in Eq. 8
    Cut the edge with the lowest T_{ij};
    $G_k^{'} = G_{k1} + G_{k2}$, $D_k = D_{k1} + D_{k2} + D_{new}$;
  End while

pattern according the size of $\{2, 3, 4\}$. We further calculate the number of groups and times periods in which people switched between groups in the sequences. We define the error function as $argmin\, \varepsilon_s + \varepsilon_g$, where ε_s is target switching error and ε_g is the group number error $(N_G - N_{gt})^2$. N_{gt} represents the number of tracks in GT. We search optimal parameters for the large group with the lowest switching error ε_s. The error function essentially guarantees that the learned group pattern is universal and movement is stable. To find an optimal set of parameters, we solve it with gradient descent, starting from multiple random initializations. Fig. 5 summarizes eight types of typical topology pattern, which are the most frequent configurations identified in group sizes, ranging from two to four.

V. GST for Tracking

In this section, we present an RGB-D MPT method based on the proposed GST model, including identifying the same groups in the sequence and associating individuals within groups.

A. Group Tracking

Typically, a group is easier to track than an individual object. The whole group usually occupies a larger region than a single person, which means that groups are less likely to get lost or drift. This observation is exploited in our framework, where the group configuration at one-time step can be used as a reference for the next one. Some grouping results are interrupted by false object detection results. Hence, we link the groups of the same topology through the time span of the 'poor detection'. Recall that a group is modeled as a set of nodes and edges in a graph G_k. We use the consistency property in Eq. 9 to estimate the virtual center state of a group as

$$\hat{g}_k^{f+F} = g_k^f + \sum_{n_i \in G_k}^{N} (Bl_i^f) + \Gamma\varphi^f, \tag{11}$$

where

$$B = diag(B_1, B_1),$$
$$B_1 = \begin{pmatrix} 0 & F/N_k \\ 0 & 0 \end{pmatrix},$$

g_k and $\hat{g}_k$ are the center and virtual center of the group, respectively. B and Γ are the parameters controlling the status variation of the group, where B defines the motions of all members in G_k, and Γ controls the noise update. F is the time interval between two groups in the sequence. We measure the motion affinity of two groups by the motion smoothness between the group mean trajectories of the two corresponding groups $\hat{g}_i^{f+F}$ and g_j^{f+F}.

B. Individual Tracking

Individual persons in the group should also be identified, when we obtain the group tracking results. Given the group relation $G = (\{n_1, \cdots, n_N\}, E)$, we adopt a linear programming framework [41] to solve the in-group association problem. Compared with the global group association, the in-group matching is a subgraph searching problem in the grouping context. We define a binary indicator matrix M_{ij} to describe the association relation among the tracklets, which decides whether the tracklets n_i and n_j belong to the same person or not. If connected, $M_{ij} = 1$; otherwise, $M_{ij} = 0$. The optimal tracklets connections in the groups are solved by

$$\arg\max_M \sum_{i,j} (A_{ij} + \Psi_{ij}) M_{ij}, \tag{12}$$

with the constraints that $\sum_i M_{ij} \leq 1$ and $\sum_j M_{ij} \leq 1$. A_{ij} denotes the appearance affinity between the nodes n_i and n_j, detailed in Sec. IV-A. Ψ_{ij} denotes the position affinity $(\pi_i = (r_i, \theta_i))$ between node n_i and n_j, according to the group center, given as g_k.

The linear programming problem in Eq. 12 can be solved by the Hungarian algorithm [41] and the iterative approximation method [42], or can be interpreted by a network flow method and solved by the successive shortest paths [10], [11]. Here, since the grouping results provide a fix time interval for individual tracking, the computational cost is greatly reduced, compared with the global association methods. We adopt the Hungarian algorithm to solve Eq. 12 in polynomial time.

C. Joint Group-individual Occlusion Handling

The proposed model relies on a fixed set of detections as input. This has the drawback that much of the image information is discarded during the non-maxima suppression step built into any detector, potentially ignoring spatially occluded persons. The group members often occlude each other.

To address this issue, we add virtual nodes to the groups when a group does not contain any appropriate detection. Multiple persons moving together usually causes occlusion. Particularly, persons can be partially or totally occluded. In this case, even the best detector is not able to discover the persons without context information. Generally, group management identifies such two groups of different sizes as different groups. Nevertheless, in our model, the centers of two groups with different sizes could also be linked by Eq. 11. Virtual nodes $\{\hat{n}_i\}$ are added to a specific group when the size of the group is less than that it ought to be. We then, consider adding virtual nodes to such groups and inferred positions of the occluded people through topological configurations. The spatial positions of the virtual nodes are estimated as

$$\hat{l}_o^f = l_o^{f-F_o} + \sum_{n_i \in G_k}^{N} (B l_i^f) + \Gamma \varphi^f, \tag{13}$$

where F_o is the occluding time of the estimated person l_o, and $\{l_i\}$ denote the motion vectors of other members in group G_k. The definition of B and Γ are the same as those in Eq. 11.

D. Single instance consistency

When persons do not move in a group, these tracklets need to be connected as a long trajectory without the grouping information. We promote the objective function in Eq. 12 as

$$\arg\max_{M} \sum_{i,j} (A_{ij} + \hat{\Psi}_{ij} + \hat{T}_{ij}) M_{ij}, \tag{14}$$

where $\sum_i M_{ij} \leq 1$ and $\sum_j M_{ij} \leq 1$. We drop the position affinity Ψ_{ij} in Eq. 12, instead adding the estimated position affinity $\hat{\Psi}_{ij}$ and the motion affinity $\hat{T}_{ij}$. The affinity $\hat{\Psi}_{ij}$ is calculated between the start position of tracklet n_i with the estimated position of $\hat{n}_i$ as

$$\hat{\Psi}_{ij} = \mathcal{G}\left(X_i - \hat{X}_i, \sum\nolimits_X\right) \mathcal{G}\left(X_j - \hat{X}_j, \sum\nolimits_X\right), \tag{15}$$

where $\hat{X}_i$ and $\hat{X}_j$ are estimated by the individual consistency property in Eq. 9. $\mathcal{G}(\cdot)$ is the Gaussian function ranging in $[0, 1]$, where the mean value is $X_i - \hat{X}_i$ and the variation is $\sum_X$. The motion affinity in Eq. 3 is modified as $\hat{T}_{ij} = T_v(\hat{n}_i, n_j) + T_o(\hat{n}_i, n_j)$. The estimated position of node $\hat{n}_i$ is calculated by $\hat{l}_i^f = C l_i^{f-F} + \Gamma \varphi^{f-F}$, which is detailed in Eq. 9. This assignment problem in Eq. 14, similar to Eq. 12, can be solved optimally by the Hungarian algorithm in polynomial time.

VI. Experiments

We evaluate the performance of the proposed GST model against state-of-the-art methods on several challenging RGB-D and RGB datasets. We then conduct group discovery and model ablation in different complex sequences. Experimental results clearly show the benefits of utilizing social topology in the RGB-D MPT application.

TABLE II
THE PARAMETERS USED IN THE EXPERIMENTS

Parameter	α_t	α_d	α_v	α_o	l	τ	λ
RGB-D	0.4	0.2	0.2	0.2	5	0.5	1.2[m]
RGB	0.4	0.2	0.2	0.2	5	0.5	50[px]

TABLE III
TRACKING RESULT COMPARISON ON RGB-D PEOPLE DATASETS

Method	MOTA↑	MOTP↑	FP↓	FN↓	IDS↓
DP [10]	62.3%	71.1%	13.4%	38.7%	48
OC [30]	71.8%	73.7%	7.7%	20.0%	19
MHT [22]	78.0%	N/A	4.5%	16.8%	32
DSA [31]	75.8%	73.7%	7.2%	18.5%	24
LGM [7]	78.3%	75.5%	4.9%	27.7%	16
PHD [48]	75.1%	74.6%	1.5%	23.3%	7
Ours	81.2%	75.6%	3.9%	12.7%	9

Datasets and Evaluation Metrics. The proposed method is tested on three kinds of public datasets, including RGB-D indoor datasets [9], [43], RGB-D outdoor datasets [31], [44], [45], and RGB tracking datasets (MOT Benchmark [46]). We adopt the commonly used CLEAR MOT tracking metrics [46], [47] to evaluate the tracking performance. Recall and Precision (Prec.) are two basic metrics. Multi-Object Tracking Accuracy (MOTA) and Multi-Object Tracking Precision (MOTP) reflect the general performance. The ratio of correctly identified detections over the average number of ground-truth and computed detections (IDF1). Mostly Tracked (MT) and Mostly Lost (ML) scores are computed on the entire trajectories and measure how many Ground Truth (GT) trajectories are successfully tracked (tracked for at least 80%), and lost (tracked for less than 20%). Fragment (Frag.) and ID Switch (IDS) record how many times the GT trajectory is interrupted and switched by a false ID. False Positive (FP) and False Negative (FN) rates record the number of false positives and negatives (missed objects), respectively.

A. Training

Training datasets include three parts: 1) 2D MOT benchmark from the MOT training sets ($i.e.$, the available camera calibration parameters); 2) *eth* & *hotel* [19] ($i.e.$, 650 tracks including top-view, desired speed and direction are provided), and 3) SDL-Campus RGB-D datasets ($i.e.$, 500 tracks which include depth values derived from depth sensors). We set the confident tracklet threshold as $l = 10$ during the training stage. Then during optimization, we set the distance threshold parameter λ as the searching field $[0, 3]$. An overview of typical settings for all parameters is provided in Table II. These parameters have been chosen conservatively and are not specified for any particular dataset. This demonstrates the robustness of the proposed model in a variety of scenarios.

B. Results on RGB-D Indoor Datasets

We apply our tracker to a publicly available dataset, the RGB-D people dataset [9], [43], which contains a sequence

of over 3000 RGB-D frames captured through three vertically mounted Microsoft Kinect sensors. The sensor configuration is placed in a busy university hall at approximately $1m$ height. For all the compared trackers in Table III, we use the same tracking input generated by a people detector [30], that is available in the Point Cloud Library (PCL 1.7) [49]. Table II shows the parameters we used in the experiments. In order to verify the accuracy and efficiency of the proposed method, we compare outputs with the following state-of-the-art methods:

- DP [10]: RGB + Dynamic Programming;
- OC [30]: RGB-D + Online Clustering;
- MHT [22]: RGB-D + Multi-model Hypothesis Tracking;
- DSA [31]: RGB-D + Depth Structure;
- LGM [7]: RGB-D + Layered Graphical Model;
- PHD [48]: RGB-D + Flow Network.

DP [10] is a dynamic programming method based on a globally-optimal but greedy solution in the RGB domain. It is not technically a grouping method, and all tracklets are considered individuals during whole tracking. OC [30] is an online grouping method based on clustering, assuming a fixed number of groups in the scene which is different from our proposed method. Our GST model is able to add and remove tracking targets according to the social affinity matrix. MHT [22] is an RGB-D tracking model based multiple hypotheses. A very large amount of hypotheses are added in the tracking process with the number of targets increasing, so the solution is more complex. DSA [31] and LGM [7] are online RGB-D grouping methods based on depth structure and layered graph, respectively. For both these methods, depth information is considered a strong constraint in the grouping, but not in the social context. Consequently, there is presently no flexible way to add and remove members in evolving groups. PHD [48] is an RGB-D data association method based on multiple probabilistic hypotheses. Different from the MHT method, PHD adopts a global network flow solution, which can avoid the hypothesis explosion problem observed with the MHT method.

Table III shows the experimental results of our approach and other state-of-the-art methods by comparison. DP is a baseline method in our experiments. We observe that our model achieves the best MOTA, MOTP, and FN, and at the same time, the second-best FP and IDS metrics. Our method improves overall performance (MOTA) by approximately 3%. We also observe that the tracklet-level association methods [10], [22], [30] usually fill small gaps between correct tracklet pairs in occlusion cases, but only during grouping [7], [31], which adds virtual nodes to the group context and can provide better estimates for the occluded objects, especially in long-term occlusion cases. The methods which do not implement grouping skills [10], [30], [48] usually fail in challenging long trajectory cases, causing fragmented trajectories in the tracking results.

C. Results on RGB-D Outdoor Datasets

We further test the proposed model on three RGB-D outdoor datasets: the ISR-sync dataset [45], the SDL dataset [31], and the LIPD dataset [44]. Each video sequence has a variable number of target objects (Car, Pedestrian, and Cyclist). The videos are recorded at 10 FPS although these datasets are very challenging since 1) the scenes are crowded (with occlusion and clutter); 2) the camera is not stationary; and 3) target objects appear in arbitrary locations with varying sizes. To keep the same tracking input with other trackers, we adopt the same detection results as in [7], [31].

TABLE IV
TRACKING RESULT COMPARISONS ON RGB-D OUTDOOR DATASETS

Dataset	Method	Recall↑	Prec.↑	MT↑	ML↓	IDS↓	Frag.↓
ISR-Sync	DP [11]	67.4	72.1	15.2	31.8	287	378
	SSP [10]	69.6	72.8	9.0	24.2	345	323
	DCO-X [1]	73.4	78.3	19.6	19.6	89	125
	SegTrack [50]	76.2	79.2	25.8	16.7	102	147
	DEEPCC [2]	79.3	87.4	27.2	19.6	134	153
	DSA [31]	82.7	86.8	28.8	13.7	90	108
	LGM [7]	85.0	89.7	27.2	19.1	121	97
	Ours	87.5	92.3	31.8	21.2	102	134
SDL	DP [10]	64.6	71.3	14.1	25.0	154	166
	SSP [11]	62.9	70.5	9.8	30.4	168	189
	DCO-X [1]	70.4	76.4	19.6	25.0	65	74
	SegTrack [50]	72.3	77.8	18.4	10.8	55	71
	DEEPCC [2]	76.5	82.6	21.2	12.4	87	95
	DSA [31]	79.6	87.3	25.0	15.2	60	68
	LGM [7]	82.4	87.3	27.6	8.7	65	57
	Ours	84.6	89.0	30.4	9.8	58	71
LIPD	DP [10]	73.2	77.8	18.1	28.6	305	198
	SSP [11]	72.8	76.4	10.4	33.8	324	219
	DCO-X [1]	78.4	78.6	19.5	22.1	92	123
	SegTrack [50]	77.6	80.2	13.0	19.5	75	118
	DEEPCC [2]	81.3	86.6	26.8	13.3	74	101
	DSA [31]	79.6	87.3	25.0	15.2	60	68
	LGM [7]	84.9	88.3	29.7	11.9	82	47
	Ours	86.7	90.0	33.8	10.4	71	65

Table IV shows the results of our model in comparison with state-of-the-art trackers: DP [10], SSP [11], DCO-X [1], SegTrack [50], DEEPCC [2], DSA [31], and LGM [7], where the DP, DSA, and LGM methods are kept the same as those adopted in the previous RGB-D people dataset. SSP is a typical data association method searching successive shortest paths in an association graph. DCO-X is an energy-based model proposed in [1], where discrete- and continuous-energy minimization is defined according to the tracklets exclusion procedure. The SegTrack method uses joint segmentation and tracking superpixels and targets within the RGB domain. The DEEPCC tracker adopts CNN-based features to design a more accurate appearance affinity. Here we use the one-camera tracker and from the statistical outputs on the RGB-D outdoor datasets reported in Table IV, we obtain similar findings to those obtained from the RGB-D indoor datasets. Compared with RGB-based trackers [2], [10], [11], [50], our method performs more accurately with more robust results (improving $\sim 7\%$ in Recall and $\sim 6\%$ in Precision), and at the same time, achieving almost all the best performance in the metrics, MT, PL, ML. This verifies that the RGB-D based model combining depth information shows a more discriminating ability than the RGB-based pixel-level measurement during spatial object association. This is integrated into our GST model as a social context constraint, showing superior performance over RGB-based technologies. In comparison with the RGB-D based tracker [7], [31], our model improves on average more than

2% in Recall and Precision on the ISR-Sync, SDL, and LIPD datasets. This confirms that our GST model with the initializing, merging, and splitting modules is more flexible at describing and dividing groups than pure grouping methods [7], [31] during tracking.

D. Results on RGB Datasets

To further assess the performance of the proposed model, we apply it to commonly used RGB datasets: the MOT Benchmark [46]. It consists of several kinds of evaluations. To validate the effectiveness and robustness in the 3D and 2D domain, here we choose the 3D MOT 2015 dataset, 2D MOT 2015 dataset, MOT16 dataset, and MOT17 dataset in the benchmark. The evaluation results in Table V are generated from the MOTChallenge benchmark website [46]. The appearance feature used in RGB datasets is different from that in the RGB-D datasets. For a fair comparison, we keep the appearance feature extraction the same with that in the MTEV method [23].

The 3D MOT 2015 dataset consists of 2 sequences for training and 2 for testing, which is part of the 2D MOT 2015 dataset. In this 3D dataset, a pedestrian's 3D position is typically obtained by projecting the 2D position of the feet of the person into the 3D world, $e.g.$, by using a homography between the image plane and the ground plane. The bottom center point of the bounding box is chosen to represent the position of the feet of the pedestrian. But by the nature of projective geometry, even slight 2D misplacements can cause large 3D errors. The sequences with a moving camera show that these errors are too large for tracking purposes, and therefore those sequences with a moving camera are not included in the 3D MOT dataset. In the task, the calibration files are used to compute a 2D homography between the image plane and the ground plane. All y coordinates are set to 0, indicating the position of the feet of the pedestrian. Correct detection requires at most $1m$ offset in position. The 2D MOT 2015 dataset consists of 11 sequences for training and 11 for testing, with a total of $11,286$ frames ($\sim$16.5 minutes) with varying FPS. Some of the videos are recorded using a mobile platform and the others are from surveillance videos. We choose two difficult sequences: AVG-TownCentre and PETS09-S2L2 sequences for test. Moreover, to validate the effectiveness of our model in the challenge sequences, we apply our GST model on the MOT16-03, MOT16-07, MOT17-08, MOT17-14 sequences in MOT16 and MOT17 datasets, which are the most crowd scenes in these two datasets. The density of sequences varies from 24.6 to 69.7.

Evaluation on 3D MOT 2015 dataset. Table V summarizes the tracking results of the proposed GST model and other state-of-the-art methods on 3D MOT 2015. We compare our model with several state-of-the-art trackers: AMIR3D [25], DBN [51], MOANA [52], LPSFM [21], and MTEV [23]. It is observed that our model outperforms the other trackers in most metrics.

The DBN tracker estimates the pedestrians and unknown targets in a probabilistic framework integrated to a dynamic model, in which a Random Forest-algorithm based classifier is capable of being trained incrementally so that new training samples can be incorporated. The LPSFM tracker models the social force as linear programming and solves it in a network flow framework. Our previous MTEV tracker is an online data association method integrating the social force in an energy formulation. Both MTEV and our model infer the position estimations of targets in a grouping manner. But our method adopts a graphical social constraint to formulate the group's member, providing a flexible way to initial, merge, and remove the members in a social context. Consequently, our results achieve about 3% improvement in overall performance MOTA.

We observe that the proposed approach shows lower group detection errors compared to the LPSFM and MTEV methods. It is because our model tries to maintain the individual and group labels consistently during the sequence by employing a dynamic topological graph that in some cases maybe fail to predict adequately the long-term occluded individuals. But in most cases, it is robust to the short-term occlusion case, as shown in Fig. 8. When the person #2 gets out of the occlusion in frame 156, our tracker assigns the label #5 to it (in the last row), but the MOANA tracker can maintain the label by the ReID technique. The advantage of our tracker is recording the group topology by the group tracking, so we can observe that the group also consists of 4 members after getting out of occlusion, the other 3 trackers miss the occluded person (row 1-3, frames 156 and 164).

Evaluation on 2D MOT dataset. In the result of experiments, we observe that the proposed model is also applicable to the 2D dataset, though real-world distances is not available. The proposed tracker is compared with other state-of-the-art trackers, including: motion segmentation and clustering (MSC) based tracker (NOMT [53], JointMC [35], and JBNOT [56]), CNN-based tracker (AM [24], AMIR2D [25] and AP-RCNN [54]), ReID based tracker (TBW [57]) and a social force (SF) based tracker (MTEV [23]). To maintain consistency with reported numbers, we follow the exact same evaluation protocol as all other approaches and use detection results provided by the MOT website, as inputs.

Table V presents tracking results on challenging sequences identified in the 2D MOT 2015 and MOT17 datasets. S-ince pixel measurements are not entirely accurate, affinity discrimination ($i.e.$, distance, orientation, and speed) among tracklets is weakened, particularly, when targets are far from the viewpoint. Despite this fundamental weakness, we achieve competitive results. For example, in the 2D image domain, CNNs-based methods show potential in appearance matching. The AM, AMIR2D, AP-RCNN, TBW, and FAMNet meth-ods are all CNNs-based methods. Therefore, leverage deep learning features can also achieve high matching rates with data association. AM also uses CNN features and an attention mechanism to track long trajectories, which, as reflected by IDS errors, prevent track switches. Additionally, AMIR2D presents an RNN structure which jointly reasons CNN-based appearance, motion, and interactional cues over a temporal window. In FAMNet designs, the network using differentiable layers can be optimized jointly to learn discriminating features and higher-order affinity models for tracking. TBW leverages a novel bounding box regression method to predict target

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3049397, IEEE Transactions on Circuits and Systems for Video Technology

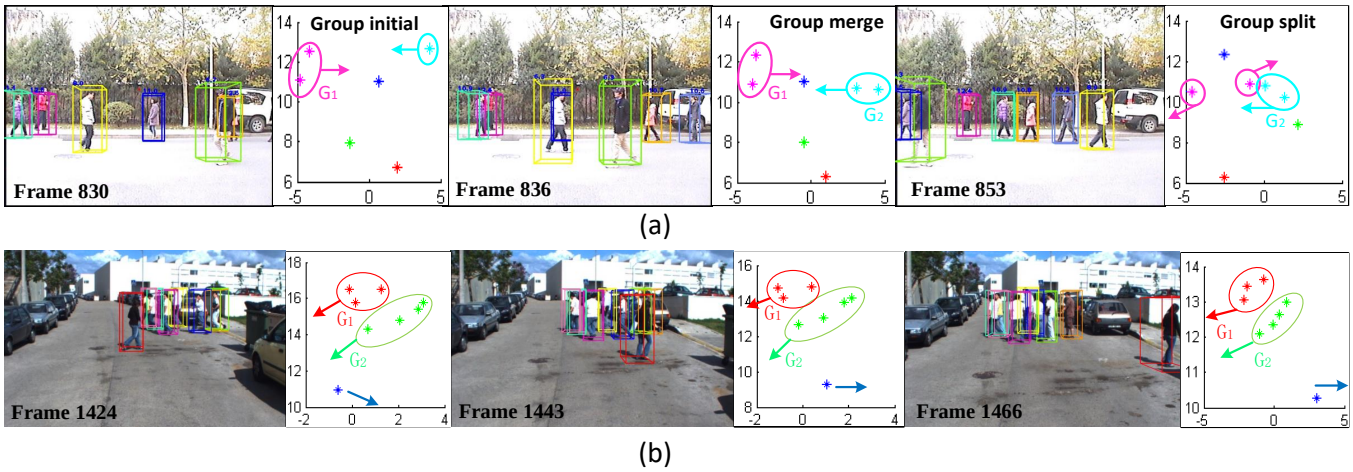


Fig. 7. Tracking results of our approach. (a) shows the tracking results of SDL datasets, and (b) shows the tracking results of ISR-sync dataset. The results contain the group birth, split, and merge events. We visualize the results in depth domain of these two datasets as well.

motion, by combining the ReID and camera motion compensation.

We observe that the CNN-based methods achieve the best MOTA results. However, in crowded scenes, *e.g.*, AVG-TownCentre and PETS09-S2L2 sequences, these methods lose the advantage in appearance discrimination. The reason for this perhaps lies in that occlusions, false detections, and missed detections deteriorate appearance matching capabilities. It is also observed that the ReID technique can improve tracking accuracy because occlusion problems occur frequently in sequences with crowded scenes. The person getting out of occlusion could be assigned the accurate ID by ReID technique. This is verified using the IDS metric in Table V.

E. Discussion

Model ablation. We systematically investigate the contribution of different components of our social affinity matrix by disabling components one at a time, then examining performance changes in terms of MOTA on the PETS09-S2L2 (3D), AVG-TownCentre (3D), 3D MOT 2015 average performance, and 2D MOT 2015 average performance. Fig. 9 provides ablation outputs. The first column is our proposed GST model with all social affinities in Eq. 3. We then, disable affinity by distance (w/o T_d), time (w/o T_t), speed (w/o T_v) and orientation (w/o T_o), sequentially. In addition, we combine speed and orientation terms into one term. $T_{v+o} = \mu\mathcal{N}(\|v_i - v_j\|) + (1-\mu)\mathcal{N}(\|o_i - o_j\|)$, where $\mathcal{N}(\cdot)$ is a min-max normalization operator ranging [0, 1], and the other two terms are kept as same as those in Eqs. 4 and 5 ($T_v + T_o$). In the experiments, we use $\mu = 0.5$.

We observe that the new social affinity T with different disabled items decreases the performance compared with the full affinity model. Time constraint appears to play the most important role in social affinity since performance decreases most on the MOTA item when this item is disabled. In other words, when two tracklets do not appear and disappear with simultaneous timestamps, they generally could not be divided to a specific group. Conversely, if they appear and disappear at

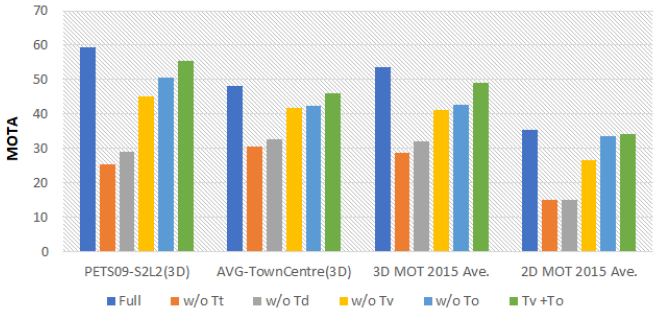


Fig. 9. Model ablation study on the 3D and 2D MOT datasets.

the same time gap, even the distance between them is longer than the social distance we defined, that is the MOTA item increases, compared with the 'w/o T_d'. This grouping result may be inaccurate from a theoretical grouping point of view; however, it is accurate for the final individual tracking results. This grouping result is also able to provide a topological reference according to graphical configuration. Their positions could be topological support for one another while moving but only if they follow the same patterns. Compared to our full tracking model including four affinities, the 'w/o T_v' and 'w/o T_o' model generate less accurate groups, some in-group members are missed and some out-group targets are added into groups, so parts of these tracks are falsely connected during individual tracking, especially the occluded members, resulting in a decreasing MOTA score. Here, the orientation constraint is more effective than the speed constraint during motion measurement which confirms that pedestrians in crowds, with the same geometrical goal always perform the same motion pattern [16], [17].

Parameter study. To examine the influence of parameters (l, τ, and λ) of the GST model, we run the tracking model and modify the corresponding parameter while keeping other parameters fixed. In Fig. 10, of each term, the relative change in MOTA performance is plotted against the parameter value. During the experiments, we run the proposed model on RGB-

TABLE V
TRACKING RESULT COMPARISONS ON TWO RGB DATASETS IN 3D/2D MOT CHALLENGE [46] WITH PUBLIC DETECTIONS

Dataset	Sequence	Method	Setting	MOTA ↑	MOTP ↑	IDF1 ↑	Density	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓	Frag. ↓
3D MOT 2015	PETS09-S2L2	AMIR3D [25]	2D+RNN	32.2	56.2	–	22.1	2.4%	4.8%	913	4,729	891	1,018
		DBN [51]	3D	57.6	63.6	–	22.1	28.6%	4.8%	805	3,049	231	245
		MOANA [52]	3D+ReID	57.6	57.0	61.3	22.1	40.5%	7.1%	1,453	2,531	107	386
		LPSFM [21]	2D+SF	41.3	55.7	–	22.1	7.1%	16.7%	640	4,776	243	271
		MTEV [23]	3D+SF	55.5	61.3	59.3	22.1	21.4%	4.8%	971	3,735	206	221
		Ours(3D)	3D+SF	58.7	64.8	62.2	22.1	38.1%	4.8%	791	2,673	112	189
	AVG-TownCentre	AMIR3D [25]	2D+RNN	15.3	54.6	–	15.9	3.1%	31.9%	1,125	4,355	571	629
		DBN [51]	3D	42.4	57.1	–	15.9	28.8%	20.4%	1,272	2,697	149	173
		MOANA [52]	3D+ReID	46.1	55.1	64.0	15.9	26.1%	24.8%	773	3,020	60	200
		LPSFM [21]	2D+SF	28.7	51.9	–	15.9	15.0%	22.6%	1,391	3,430	277	330
		MTEV [23]	3D+SF	43.2	57.2	62.3	15.9	22.1%	24.8%	938	3,380	253	229
		Ours(3D)	3D+SF	46.7	58.1	66.1	15.9	29.6%	22.6%	812	2,845	181	192
	Average Performance	AMIR3D [25]	2D+RNN	25.0	55.6	–	–	3.0%	27.6%	2,038	9,084	1,462	1,647
		DBN [51]	3D	51.1	61.0	–	–	28.7%	17.9%	2,077	5,746	380	418
		MOANA [52]	3D+ReID	52.7	56.3	62.4	–	28.4%	22.0%	2,226	5,551	167	586
		LPSFM [21]	2D+SF	35.9	54.0	–	–	13.8%	21.6%	2,031	8,206	520	601
		MTEV [23]	3D+SF	49.8	62.2	–	–	25.7%	17.2%	1,909	7,115	459	450
		Ours(3D)	3D+SF	53.8	61.4	–	–	32.1%	16.8%	1,633	5,618	243	381
2D MOT 2015	PETS09-S2L2	NOMT [53]	2D+MSC	53.4	70.5	43.6	22.1	14.3%	9.5%	884	3,465	142	208
		JointMC [35]	2D+MSC	56.0	71.4	41.1	22.1	23.8%	4.8%	942	3,162	142	220
		AM [24]	2D+CNN	47.7	69.2	44.3	22.1	16.7%	14.3%	718	4,206	115	356
		AMIR2D [25]	2D+RNN	47.0	70.5	36.3	22.1	11.9%	9.5%	616	4,236	254	397
		AP-RCNN [54]	2D+CNN	38.9	70.8	34.3	22.1	2.4%	9.5%	552	5,164	179	328
		MTEV [23]	2D+SF	51.8	70.4	42.6	22.1	16.7%	11.9%	715	3,812	172	161
		Ours(2D)	2D+SF	53.8	70.9	43.9	22.1	18.1%	7.1%	712	3,542	169	229
	AVG-TownCentre	NOMT [53]	2D+MSC	31.6	70.1	44.6	15.9	11.1%	36.3%	681	4,060	146	233
		JointMC [35]	2D+MSC	43.1	69.8	62.2	15.9	29.2%	32.3%	922	3,116	28	213
		AM [24]	2D+CNN	37.5	68.1	53.9	15.9	14.2%	30.5%	645	3,742	79	332
		AMIR2D [25]	2D+RNN	36.2	69.5	52.5	15.9	26.1%	17.7%	1,448	2,882	234	389
		AP-RCNN [54]	2D+CNN	28.4	66.9	44.7	15.9	4.0%	27.9%	941	4,005	169	412
		MTEV [23]	2D+SF	33.7	70.2	45.1	15.9	22.1%	30.1%	942	3,756	113	163
		Ours(2D)	2D+SF	37.2	70.4	57.6	15.9	27.0%	27.9%	841	3,619	128	226
	Average Performance	NOMT [53]	2D+MSC	33.7	71.9	44.6	–	12.2%	44.0%	7,762	32,547	442	823
		JointMC [35]	2D+MSC	35.6	71.9	45.1	–	23.2%	39.3%	10,580	28,508	457	969
		AM [24]	2D+CNN	34.3	70.5	48.3	–	11.4%	43.4%	5,154	34,848	348	1,463
		AP-RCNN [54]	2D+CNN	38.5	72.6	47.1	–	8.7%	37.4%	4,005	33,203	586	1,263
		AMIR2D [25]	2D+RNN	37.6	71.7	46.0	–	15.8%	26.8%	7,933	29,397	1,026	2,024
		MTEV [23]	2D+SF	33.8	71.1	44.8	–	12.1%	34.8%	9,232	31,743	722	1,257
		Ours(2D)	2D+SF	35.1	71.7	47.3	–	17.7%	36.6%	6,874	31,623	592	1,338
MOT17 (2D)	Average performance	FAMNet [55]	2D+CNN	52.0	76.5	48.7	–	19.1%	33.4%	14,138	253,616	3,072	5,318
		JBNOT [56]	2D+MSC	52.6	77.1	50.8	–	19.7%	35.8%	31,572	232,659	3,050	3,792
		TBW [57]	2D+CNN	53.5	78.0	52.3	–	19.5%	36.6%	12,201	248,047	2,072	4,611
		MTEV [23]	2D+SF	49.7	77.1	50.4	–	17.0%	36.7%	21,811	258,641	3,077	4,339
		Ours(2D)	2D+SF	52.7	77.5	53.7	–	20.4%	38.1%	27,032	237,539	1,703	3,147

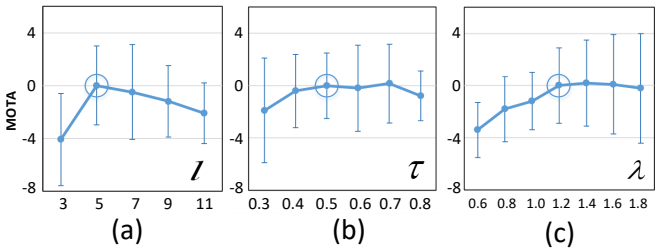


Fig. 10. The influence of parameters l, τ, λ on the RGB-D Indoor and Outdoor datasets.

D indoor and RGB-D outdoor datasets, the average mean-normalized value is shown along with error bars, depicting variation between various sequences. The results shown here are averaged across all datasets and normalized for enhanced readability. The error bars represent standard deviation around the mean. The parameter value used in the experiments is marked with a circle. As can be seen, the choice of parameters is rather conservative and does not correspond to the best parameter setting, when compared with those in Table II. However, this could also be an indication that the proposed model is not over-tuned on the test datasets.

Error pattern analysis. In model training, we summarize eight typical topology patterns, as shown in Fig. 5, which provide evidence of grouping in the initial stages. Besides the eight typical patterns, there are other usual patterns in the training sets, such as a line-like topology pattern, shown in the G_3 of Fig. 1. This topology is not added to the typical topology patterns, since it is not a compact structure and does not satisfy *Constraint II* in Eq. 8. In experiments, we investigate the influence of an unstable topology pattern. Fig. 11 provides a visual representation of two groups consisting of four members. The first group is encoded by the topology pattern T_8 in Fig. 5. The second, is a line-like pattern and if this pattern is adopted as a typical pattern in grouping, the formed group would have easily split and merged, compared to topology T_8. Occlusion handling may consider the object

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2021.3049397, IEEE Transactions on Circuits and Systems for Video Technology

Fig. 8. Tracking results in AVG-TownCentre sequence of 3D MOT 2015 dataset. The results from (a)-(d) belong to MOANA [52], DBN [51], LPFSM [21], and Ours (3D) respectively. (e) visualizes the group topology evolving of our method.

Fig. 11. The influence of error pattern in the MOT17-03 sequence.

in the case of occlusion, but it has already actually split from the group, which creates IDS trajectory errors in tracking. In addition, this increases the burden of group management and it is observed that the first group has a stable pattern in Fig. 11.

Time analysis. The number of persons greatly affect the computational time. We run experiments on an Intel 3.8GHz PC with 8G memory, and implement the codes in Matlab. Without code optimization, it achieves a tracking speed of ∼20 Hz when there are on average 10 persons to be tracked. The target number increases to 30, it achieves an average speed of ∼7 Hz. When applied to 3D MOT 2015 dataset, calculating one frame takes approximately 82ms on average. The results

TABLE VI
TIME COMPARISON AT DIFFERENT STEPS OF THE PROPOSED METHOD. RESULTS ARE REPORTED WITH THE 3D MOT 2015 DATASET

Component	TG	STM	GT	IT	JT	ST	Full
Time	5ms	9ms	39ms	21ms	5ms	3ms	82 ms

are provided in Table VI. Our method can be divided into six steps, including:

- TG: Tracklets generation;
- STM: Social topology matrix calculating;
- GT: Group tracking;
- IT: Individual tracking;
- JT: Joint group-individual tracking;
- ST: Single object tracking.

Note that the computational time for object detection is not included, but the tracklets generation, appearance, and motion feature extraction are included in the above estimates of computational time. The most time-consuming part happens in the group tracking and individuals tracking (including the group online updating and occluded object estimation).

From a theoretical perspective, the optimization of the probabilistic graph-based methods [7], [33] or the energy-based methods [1] are non-convex optimization problems. To compute the gradient, an alternative approach involving the Hungarian algorithm and K-means clustering is applied. Such

clustering needs multiple initial starts to reach a reasonable local optimum, which leads to high computational cost. Some RGB trackers are based on a complex motion context, the motion constraint is inferred by superpixels [50], and optical flow [58], taking a large amount of time. Our solution, in contrary, has a closed-form solution based only on the deterministic Hungarian algorithm and thus can be computed much more efficiently.

VII. Conclusion

We proposed the Graphical Social Topology model solve the multi-person tracking problem in a joint group modeling and MPT framework. The dynamics of moving objects were formulated in a developed graph representation. We learned the typical topology configurations in training datasets and implemented these trained topology patterns to infer the group structure and dynamics combining them with a social topology matrix. Meanwhile, we solved the self-occlusion problem in the topology update and identified the individual objects after grouping. Experiments on both RGB-D and RGB datasets showed that our graphical topology approach significantly improved the MPT performance, validating that the group-level constraint is effective for tracking in crowded scenes.

In future work, the proposed framework has several aspects that could be extended. The group training process would require more extensive annotated datasets, providing a more accurate group inference. A CNN-based appearance descriptor could improve the individual tracking performance. Our approach is general enough to allow the embedding of these methods in the presented framework separately and in our modular fashion, towards an end-to-end RGB-D application.

References

[1] A. Milan, K. Schindler, and S. Roth, "Multi-target tracking by discrete-continuous energy minimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2054–2068, 2016.

[2] E. Ristani and C. Tomasi, "Features for multi-target multi-camera tracking and re-identification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6036–6046.

[3] J. Xiang, G. Xu, C. Ma, and J. Hou, "End-to-end learning deep crf models for multi-object tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2020.

[4] R. Yu, I. Cheng, B. Zhu, S. Bedmutha, and A. Basu, "Adaptive resolution optimization and tracklet reliability assessment for efficient multi-object tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 7, pp. 1623–1633, 2018.

[5] H. Zhou, W. Ouyang, J. Cheng, X. Wang, and H. Li, "Deep continuous conditional random fields with asymmetric inter-object constraints for online multi-object tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 4, pp. 1011–1022, 2019.

[6] A. Ess, B. Leibe, K. Schindler, and L. Van Gool, "Robust multiperson tracking from a mobile platform," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 10, pp. 1831–1846, 2009.

[7] S. Gao, Z. Han, C. Li, Q. Ye, and J. Jiao, "Real-time multipedestrian tracking in traffic scenes via an RGB-D-based layered graph model," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2814–2825, 2015.

[8] T. Linder, S. Breuers, B. Leibe, and O. A. Kai, "On multi-modal people tracking from mobile platforms in very crowded and dynamic environments," in *IEEE International Conference on Robotics and Automation*, 2016, pp. 5512–5519.

[9] M. Luber, L. Spinello, and O. A. Kai, "People tracking in RGB-D data with on-line boosted target models," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 3844–3849.

[10] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1201–1208.

[11] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua, "Multiple object tracking using k-shortest paths optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1806–1819, 2011.

[12] A. Ess, B. Leibe, and L. Van Gool, "Depth and appearance for mobile scene analysis," in *IEEE International Conference on Computer Vision*, 2007, pp. 1–8.

[13] Z. Cai, J. Han, L. Liu, and L. Shao, "RGB-D datasets using microsoft kinect or similar sensors: a survey," *Multimedia Tools and Applications*, vol. 76, no. 3, pp. 4313–4355, 2017.

[14] R. Layne, S. Hannuna, M. Camplani, J. Hall, T. Hospedales, T. Xiang, M. Mirmehdi, and D. Damen, "A dataset for persistent multi-target multi-camera tracking in RGB-D," pp. 2160–7516, 2017.

[15] C. Premebida, J. Carreira, J. Batista, and U. Nunes, "Pedestrian detection combining RGB and dense LIDAR data," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 4112–4117.

[16] M. Moussaïd, N. Perozo, S. Garnier, D. Helbing, and G. Theraulaz, "The walking behaviour of pedestrian social groups and its impact on crowd dynamics," *PloS one*, vol. 5, no. 4, p. e10047, 2010.

[17] H. Singh, R. Arter, L. Dodd, P. Langston, E. Lester, and J. Drury, "Modelling subgroup behaviour in crowd dynamics DEM simulation," *Applied Mathematical Modelling*, vol. 33, no. 12, pp. 4408–4423, 2009.

[18] M. Cristani, G. Paggetti, A. Vinciarelli, L. Bazzani, G. Menegaz, and V. Murino, "Towards computational proxemics: Inferring social relations from interpersonal distances," in *IEEE Third International Conference on Privacy*, 2011, pp. 290–297.

[19] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *IEEE International Conference on Computer Vision*, 2009, pp. 261–268.

[20] Z. Qin and C. R. Shelton, "Improving multi-target tracking via social grouping," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1972–1978.

[21] L. Leal-Taixé, G. Pons-Moll, and B. Rosenhahn, "Everybody needs somebody: Modeling social and grouping behavior on a linear programming multi-people tracker," in *IEEE International Conference on Computer Vision Workshops*, 2011, pp. 120–127.

[22] T. Linder and O. A. Kai, "Multi-model hypothesis tracking of groups of people in RGB-D data," in *International Conference on Information Fusion*, 2014, pp. 1–7.

[23] S. Gao, Q. Ye, J. Xing, A. Kuijper, Z. Han, J. Jiao, and X. Ji, "Beyond group: Multiple person tracking via minimal topology-energy-variation," *IEEE Transactions on Image Processing*, vol. 26, no. 12, pp. 5575–5589, 2017.

[24] Q. Chu, W. Ouyang, H. Li, X. Wang, B. Liu, and N. Yu, "Online multi-object tracking using cnn-based single object tracker with spatial-temporal attention mechanism," in *IEEE International Conference on Computer Vision*, 2017, pp. 4846–4855.

[25] A. Sadeghian, A. Alahi, and S. Savarese, "Tracking the untrackable: Learning to track multiple cues with long-term dependencies," in *IEEE International Conference on Computer Vision*, 2017, pp. 300–311.

[26] K. Hilke, H. Wolfgang, and A. Michael, "Joint detection and online multi-object tracking," in *IEEE Computer Vision and Pattern Recognition Workshop*, 2018, pp. 1540–1548.

[27] S. H. Bae and K. J. Yoon, "Confidence-based data association and discriminative deep appearance learning for robust online multi-object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1–1, 2018.

[28] P. Chu, H. Fan, C. C. Tan, and H. Ling, "Online multi-object tracking with instance-aware tracker and dynamic model refreshment," in *IEEE Winter Conference on Applications of Computer Vision*, 2019, pp. 161–170.

[29] S. Song and J. Xiao, "Tracking revisited using RGBD camera: Unified benchmark and baselines," in *IEEE International Conference on Computer Vision*, 2013, pp. 233–240.

[30] M. Munaro, F. Basso, and E. Menegatti, "Tracking people within groups with RGB-D data," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 2101–2107.

[31] S. Gao, Z. Han, D. Doermann, and J. Jiao, "Depth structure association for RGB-D multi-target tracking," in *IEEE International Conference on Pattern Recognition*, 2014, pp. 4152–4157.

[32] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, pp. 4282–4286, 1995.

[33] Z. Qin and C. Shelton, "Social grouping for multi-target tracking and head pose estimation in video." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2082–2095, 2016.

[34] L. Bazzani, M. Zanotto, M. Cristani, and V. Murino, "Joint individual-group modeling for tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 4, pp. 746–759, 2015.

[35] K. Margret, T. Siyu, A. Bjorn, B. Thomas, and S. Bernt, "Motion segmentation and multiple object tracking by correlation co-clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2018.

[36] Z. Han, J. Jiao, B. Zhang, Q. Ye, and J. Liu, "Visual object tracking via sample-based adaptive sparse representation (AdaSR)," *Pattern Recognition*, vol. 44, no. 9, pp. 2170–2183, 2011.

[37] L. E. Navarroserment, C. Mertz, and M. Hebert, "Pedestrian detection and tracking using three-dimensional ladar data," *International Journal of Robotics Research*, vol. 29, no. 12, pp. 1516–1528, 2010.

[38] P. Dollr, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1532–1545, 2014.

[39] F.-Y. Wu, "The potts model," *Review of Modern Physics*, vol. 54, no. 1, pp. 235–268, 1982.

[40] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. part i: Dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, 2004.

[41] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1, pp. 83–97, 1955.

[42] R. T. Collins, "Multitarget data association with higher-order motion models," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1744–1751.

[43] L. Spinello and K. O. Arras, "People detection in rgb-d data," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 3838–3843.

[44] "LIPD dataset in urban environment," http://www2.isr.uc.pt/ cpremebida/dataset.

[45] L. Oliveira, U. Nunes, P. Peixoto, M. Silva, and F. Moita, "Semantic fusion of laser and vision in pedestrian detection," *Pattern Recognition*, vol. 43, no. 10, pp. 3648–3659, 2010.

[46] "Multiple object tracking benchmark," http://motchallenge.net.

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

[48] N. Wojke and D. Paulus, "Global data association for the probability hypothesis density filter using network flows," in *IEEE International Conference on Robotics and Automation*, 2016, pp. 567–572.

[49] R. B. Rusu and S. Cousins, "3D is here: Point cloud library (PCL)," in *IEEE International Conference on Robotics and Automation*, 2011, pp. 1–4.

[50] A. Milan, L. Leal-Taix, K. Schindler, and I. Reid, "Joint tracking and segmentation of multiple targets," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5397–5406.

[51] T. Klinger, F. Rottensteiner, and C. Heipke, "Probabilistic multi-person tracking using dynamic bayes networks," in *International Society for Photogrammetry and Remote Sensing on Image Sequence Analysis*, 2015, pp. 435–442.

[52] Z. Tang, R. Gu, and J. Hwang, "Joint multi-view people tracking and pose estimation for 3d scene reconstruction," in *IEEE International Conference on Multimedia and Expo*, 2018, pp. 1–6.

[53] W. Choi, "Near-online multi-target tracking with aggregated local flow descriptor," in *IEEE International Conference on Computer Vision*, 2015, pp. 3029–3037.

[54] L. Chen, H. Ai, C. Shang, Z. Zhuang, and B. Bai, "Online multi-object tracking with convolutional neural networks," in *IEEE International Conference on Image Processing*, 2017, pp. 645–649.

[55] P. Chu and H. Ling, "Famnet: Joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking," *arXiv preprint arXiv:1904.04989*, 2019.

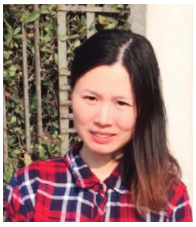
[56] R. Henschel, Y. Zou, and B. Rosenhahn, "Multiple people tracking using body and joint detections," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[57] P. Bergmann, T. Meinhardt, and L. Leal-Taix, "Tracking without bells and whistles," in *IEEE International Conference on Computer Vision*, 2019.

[58] L. Leal-Taixé, M. Fenzi, A. Kuznetsova, B. Rosenhahn, and S. Savarese, "Learning an image-based motion context for multiple people tracking," in *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, 2014, pp. 3542–3549.

Shan Gao received his B.S. degree in communication engineering from Nankai University, Tianjin, China, in 2010. He received his M.S. and Ph.D. degree from the University of the Chinese Academy of Sciences, Beijing, in 2013 and 2016. He was a postdoctoral researcher in automation department at Tsinghua University, Beijing, China in 2016. He joined the faculty at Northwestern Polytechnical University (NPU) in 2018, where he is currently an Assistant Professor with Unmanned System Research Institute. His research interests include object detection and tracking, image processing, and multi-sensor fusion.

Qixiang Ye (M'10-SM'15) received the B.S. and M.S. degrees in mechanical and electrical engineering from Harbin Institute of Technology, China, in 1999 and 2001, respectively, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences in 2006. He has been a professor with the University of Chinese Academy of Sciences since 2009, and was a visiting assistant professor with the Institute of Advanced Computer Studies (UMIACS), University of Maryland, College Park until 2013. His research interests include image processing, visual object detection and machine learning. He has published more than 50 papers in refereed conferences and journals, and received the Sony Outstanding Paper Award.



Li Liu received the Ph.D. degree in information and communication engineering from the National University of Defense Technology (NUDT), China, in 2012. She joined the faculty at NUDT in 2012, where she is currently an Associate Professor with the College of System Engineering. During her PhD study, she spent more than two years as a Visiting Student at the University of Waterloo, Canada, from 2008 to 2010. From 2015 to 2016, she spent ten months visiting the Multimedia Laboratory at the Chinese University of Hong Kong. From 2016 to 2018, she is visiting Machine Vision Group at the University of Oulu, Finland. Her current research interests include texture analysis, image classification, object detection and scene understanding.

Arjan Kuijper holds a chair in Mathematical and Applied Visual Computing at TU Darmstadt and is a member of the management of Fraunhofer IGD, responsible for scientific dissemination. He obtained a MSc. in applied mathematics from Twente University and a PhD from Utrecht University, both in the Netherlands. He was assistant research professor at the IT University of Copenhagen, Denmark, and senior researcher at RICAM in Linz, Austria. He obtained his habitation degree from TU Graz, Austria. He is author of over 250 peer-reviewed publications, and serves as reviewer for many journals and conferences, and as program committee member and organizer of conferences. His research interests cover all aspects of mathematics-based methods for computer vision, graphics, imaging, pattern recognition, interaction, and visualization.

Xiangyang Ji received the B.S. degree in materials science and the M.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 1999 and 2001, respectively, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. He joined Tsinghua University, Beijing, China, in 2008, where he is currently a Professor in the Department of Automation, School of Information Science and Technology. He has published more than 100 referred conference and journal papers. His current research interests include signal processing, image/video compression and communication, intelligent imaging.