IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 10 March 2024, accepted 26 March 2024, date of publication 29 March 2024, date of current version 10 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3383138

RESEARCH ARTICLE

Learning to Track With Dynamic Message Passing Neural Network for Multi-Camera Multi-Object Tracking

SHAVANTREVVA BILAKERI AND KARUNAKAR A. KOTEGAR, (Senior Member, IEEE)
Department of Data Science and Computer Applications, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal 576104, India
Corresponding author: Karunakar A. Kotegar (karunakar.ak@manipal.edu)

ABSTRACT Multi-camera multi-object tracking (MC-MOT) has become pivotal in various real-world applications within computer vision. Despite extensive research, solving the data association problem remains one of the most formidable challenges in MC-MOT pipeline. This challenge is compounded by factors such as varying illumination, diverse walking patterns, and trajectory occlusions. In recent years, graph neural networks (GNNs) have emerged as promising tools for enhancing data association. However, prevalent graph-based MC-MOT methods often rely on computationally inefficient min-cost flow approaches for cross-camera association, with static graph structures that lack adaptability to new detections. Moreover, these methods typically process cameras in pairs, leading to localized solutions rather than a holistic global approach. To address these limitations, we propose a two-stage lightweight cross-camera tracker designed to achieve a global solution efficiently. Our approach prioritizes the quality of local tracklets, enhancing them through supervised learning on multi-source datasets using the DeepSort model. For multi-camera association, we leverage the dynamic connectivity of Message Passing Graph Neural Networks (MPGNNs) to jointly learn features and similarities previously untapped in this domain. Our proposed model significantly improves detection accuracy and feature extraction, outperforming current MC-MOT algorithms on cross-camera benchmarks. This advancement marks a notable step forward in the field, offering more precise tracking capabilities and demonstrating the potential of integrating state-of-the-art techniques for enhanced performance in complex tracking scenarios.

INDEX TERMS Deep learning, computer vision, multi-object tracking, object detection, multi-camera multiple object tracking, graph neural network.

I. INTRODUCTION

The MC-MOT aims to deduce an entire cross-camera trajectory for each target. It has numerous implications, including crowd behavior analysis [1], [2], in-store consumer behavior analysis [3], city traffic control [4], and pedestrian monitoring [5], [6], visual tracking [7]. Many issues still need to be resolved, even though recent MC-MOT approaches have shown promising results in a number of large-scale datasets. The current studies [8], [9] typically solve the MC-MOT in two steps: (1) the local tracklet generation phase tracks each target that is detected and generates a local trajectory within a single camera, and (2) the cross-camera tracklet matching phase compares local tracklets from all the cameras to generate a full trajectory for each target across the entire multi-camera network. Due to many practical issues, such as inconsistent lighting conditions, varying object movement patterns, or the occlusions of the objects between the cameras, the data association (tracklet matching) has become a more challenging task. Data association is a vital stage in assessing the performance of an MC-MOT pipeline.

Despite years of effort, MC-MOT remains unsolved due to unknown target counts, and the complexity of predicting trajectories in multi-camera setups [10], [11]. This leads to

The associate editor coordinating the review of this manuscript and approving it for publication was Mehul S. Raval.

disruption in the relationship between an object's trajectory representation and its prior feature vectors, which results in incorrect identification (ID) assignments. For an online tracking system, failures in data association at a certain frame could have long-term negative implications. As a result, enhancing data association is vital to deciding how well an MC-MOT algorithm performs with the least model complexity that is better suited for real-time applications.

The first task of the two-stage MC-MOT approach is to generate local tracklets. However, most of the existing two-stage MC-MOT methods [12], [13], [14], [15], [16] consider the detector and feature extractor trained on single-domain (data collected in one ambient) datasets for local tracklet generation. However, a single domain offers few training samples and scene details. Despite strong performance on a seen domain (trained and tested on the same source dataset), most see a sharp fall in performance on unseen domains (trained and tested on different source datasets). The feature extractor will inevitably search for people in new scenarios in practical applications. Therefore, learning a model with excellent domain generalization (DG) capabilities is essential. The DG uses numerous datasets from various distributions, offering more training data that include a wide range of variances that helps to improve the model generalization capability. The performance of two-stage MC-MOT highly depends on the quality of local tracklets. Therefore, enhancing the quality of local tracklets is essential. Aiming at this, our study offers a completely new approach by generalizing the detector and feature extractor by training it on multi-source datasets. The existing study led us to address the following research questions, as follows:

1) By Supervising the detector on multi-source datasets improve the detector's generalization capability?
2) How does the improvement in detector quality and quantity help in improving tracking accuracy?
3) Is improving local tracklet quality contribute to improve multi camera multi object tracking accuracy?
4) Does improving the data association help in improving overall tracking performance?

Deep learning models are indeed resource-hungry, they can provide significant performance improvements with appropriate training data and computational resources. Also, tracking performance highly depends on the quality and quantity of the detections provided by the detectors. However, the existing person detection models are lacking in generalization capability. This makes room for further improvement in the detector's performance. For MC-MOT, as getting more precision detection is important, generating local trajectory from each camera is equally responsible in achieving good MC-MOT tracking accuracy. However, the conventional MC-MOT methods have considered the self-local tracker, which is tailored on single person Re-ID dataset, that lacks the diversity. Also, such trackers consider only the high score detection bounding boxes and discard the objects detected with low score, which leads to true object missing from data association even after being detected. Therefore, there is a need for two state data associations while generating local trajectories. Despite extensive research, solving the data association problem remains one of the most formidable challenges in MC-MOT pipeline. In recent years, graph neural networks (GNNs) have emerged as promising tools for enhancing data association. However, prevalent graph-based MC-MOT methods often rely on computationally inefficient min-cost flow approaches for cross-camera association, with static graph structures that lack adaptability to new detections. Moreover, these methods typically process cameras in pairs, leading to localized solutions rather than a holistic global approach. There is a need for new data association techniques which can provide dynamic global trajectory in multi-camera networks.

In single-camera MOT, the association itself is a difficult task. However, in MC-MOT environments, this effort is made considerably more complicated due to inconsistent lighting conditions as well as differences in occlusion and viewpoint across the cameras. In reality, there are other situations (appearance change/cloth change) that are even more complicated than just creating more new IDs, which could seriously impair tracking accuracy. Also, the performance of an MC-MOT algorithm is greatly influenced by cross-camera data association. We find many works focused on enhancing the cross-camera data association, such as greedy approximation [17], min-cost conventional graphs approach [18], and 3D pose estimation from multiple views [19]. Also, the assignment problem has been addressed by the majority of earlier methods using features that come from an underlying object detector, for example, by using nearest neighbors, clustering, or a case of non-negative matrix factorization, etc. In recent years, the use of GNN for prediction has gained more popularity [20], [21], [22], [23]. Specifically for MOT, recent work [24], [25], [26], [27] formulates data association as an edge classification task with GNNs, where each node denotes an object, and each edge relating to two nodes represents the similarity between detection and tracklets.

With the goal of having robust data association, our study offers the data association module built on the GNN structure for the MC-MOT task. The GNN is useful for learning features and similarities simultaneously. The concept of simultaneously learning the feature representation and similarity has already been put forth, and it has been shown to be effective in a variety of contexts, including vehicle re-identification [28], human pose recovery [29], and single-camera MOT [25], [30], [31], [32]. However, to the best of our knowledge, it has not yet been taken into account for the MC-MOT task. We treat each local tracklets as a node of the graph and leverage its appearance and spatial information at time step t. Instead of doing association by pairs, we use a Dynamic message-passing network (MPN) to execute learning directly in the graph domain and offer a single global association solution for all cameras. In comparison to state-of-the-art approaches,

this dynamic graph formulation with a generalized local tracklet model improves ID assignment by exploiting feature representation and similarity measures simultaneously.

The major contributions:

- We demonstrate that supervising it on multi-source datasets improves the performance of local tracklet generation and detection.
- For the first time, we have employed GNN into an MC-MOT domain for feature extraction and similarity measure simultaneously.
- Link prediction with dynamic MPGNN formulation is introduced as a new MC-MOT framework for cross-camera tracking.
- The MPN is integrated with the proposed dynamic graph, enabling the dynamic accumulation of spatial and temporal information to produce new graph representations, those results in extremely accurate link predictions.
- A comprehensive ablation study and comparison with cutting-edge methods demonstrate the effectiveness of the proposed technique.

II. RELATED WORKS

The convention techniques targeted for accurate, robust, and fast-tracking of multiple targets, especially as they transited across multiple cameras. Several research efforts expended over the years to effectively address various challenges while tracking persons in a multi-camera network. The existing literature offered two primary methods for solving multi-person tracking in multi-camera environments. The first approach involved the collection of detections from all the cameras and then computing the similarity among the detections to generate a complete multi-camera trajectory for each individual (global approach). The second approach was distributed/two-stage approach, in which a local trajectory from each view was generated, and then using the re-identification model, the correspondence between the local trajectory was estimated to generate a complete global trajectory.

A. GLOBAL APPROACH

Numerous studies employed a global approach to tracking, wherein all input detections were regarded as nodes in a graph, and the connections between nodes represent their level of similarity. The approach introduced by Bredereck et al. [12] involved using a greedy matching technique for single-camera tracking, followed by obtaining 3D geometric positions through triangulation between cameras to track corresponding objects across multiple cameras for improved tracking performance. In order to improve this similarity metric, an advanced feature extraction algorithm was necessary to extract the most salient features from the detections. The work presented by Chen et al. [6] utilized a Re-ID feature extraction technique to determine the edge weights, which were then subjected to a min-cut/max-flow algorithm for tracking. A different team of researchers [54] from UCF presented an algorithm for optimizing the global maximum clique called GMMCP. This algorithm calculated edge weights by considering both appearance similarity (through histogram comparison) and motion similarity (through constant velocity). Their previous work [55], which introduced the GMCP algorithm, served as the basis for this paper. The main distinction between the two algorithms was GMMCP calculated the cost function for multiple tracklet cliques at once. Interestingly, Duke University researchers also published a paper [56] in a similar manner. Although they employed the same global information association strategy as [54], they only used detection appearance features for edge weight calculation. The method proposed by Ristani and Tomasi [10] involved the use of a person detector to extract bounding box observations from video streams. Additionally, a feature extractor was employed to gather motion and appearance features from these observations, which were then transformed into correlations and labeled via correlation clustering optimization. Finally, post-processing steps were carried out to fill in any missing detections and eliminate tracks with low confidence levels.

B. TWO-STAGE APPROACH

Another set of research focused on a two-stage approach called tracklet-tracklet matching, which meant tracklets generated from every camera were matched to find a global trajectory. Following this, Xu et al. [16] proposed a Hierarchical Composition of Tracklet (HCT) framework to match local tracklets by utilizing multiple cues of targets such as appearances and their ground plane locations. The appearance features were easily influenced by the changes in illumination, pose, and viewpoint, which were most common in the multi-camera network. To build a robust model, Lee et al. [57] put forward a resilient technique for tracing a person's identity across several cameras utilizing unsupervised online learning. In this, local tracklets were generated by employing a method that utilized multi-kernel adaptive segmentation to track individuals with the assistance of local object detection, thereby generating an ideal foreground mask for feature extraction in ICT. They incorporated a color transfer method to address the issue of varying illumination in ICT. Additionally, they leveraged pose-invariant appearance features to overcome pose and camera viewpoint discrepancies between neighboring cameras, and the integration of context features boosted the performance of ICT. However, the model showed poor performance when regional color and texture features were extracted from a small area (i.e., a person visible in a small size). Xu et al. [58] hypothesized considering semantic attributes would serve as powerful cues for associating human trajectories across cameras. They employed a scene-centered spatio-temporal parsing graph which first estimated the 3D geometry of the scene from multiple camera views. This enabled them to project the camera views onto a common 3D reference frame centered at the scene. Next, spatio-temporal parsing was applied to extract human poses and movements from the

TABLE 1. Literature summary of MC-MOT Algorithms.

TrackName	Optimization method	Evaluation Benchmark	Association methods	Solution approach	Tracking method	Camera settings
GMMCP [33]	Binary Mixed Integer Linear Program	Duke MTMC and EPFL	Appearance and dynamic motion	Global	Offline	Overlapping, disjoined
MYTRACKER [34]	Maximum weighted Independent Set (MWIS)	Duke MTMC and NLPR_MCT	Appearance, motion	Two-step	online	disjointed
DyGLIP [30]	Link prediction	PETS09, CAMPUS, MCT, EPFL, and Cityflow	appearance	Two-step	online	Overlapping disjoined
TRACTA [35]	Optimal assignment problem	PETS09, CAMPUS, EPFL, MCT	Appearance, motion, camera topology, and location information	Two-step	online	Overlapping disjoined
B&P [18]	Multi-commodity network flow	PETS09	Appearance	Two-step	online	Overlapping
LAAM [36]	Graph optimization	DukeMTMC and CityFlow	appearance	Two-step	offline	Overlapping disjoined
MHT-DAM [37]	Maximum weighted Independent Set (MWIS)	MOT challenge, PETS09	Appearance and motion	Two-step	online	Overlapping
KSP [38]	Integer programming problem	PETS09	Appearance and motion	Two-step	online	Overlapping
DP [39]	Dynamic programming	PETS09	Appearance and motion	Two-step	online	Overlapping
DEEPSORT + 2WGMMF + BS [4]	Probabilistic	CityFlow	Appearance and motion	Two-step	online	Disjointed
TC + FVS + BA [4]	Weighted combination of cost	CityFlow	Appearance and motion	Two-step	offline	Disjointed
DeepCC [10]	Correlation clustering	DukeMTMC and NLPR_MCT	Appearance and motion	Two-step	offline	Disjointed
Ddashcam [40]	Non-linear	CityFlow	Appearance and motion	Two-step	offline	Disjointed
UWIPL [41]	Greedy search	CityFlow	Appearance and motion	Two-step	offline	Disjointed
TSCT + TA [42]	clustering	CityFlow	Appearance and motion	Two -step	offline	Disjointed
MCMT-TBM [43]	Non-linear	EPFL	Appearance and motion	global	online	Overlapping disjointed
MOANA + FVS + BA [4]	probabilistic	CityFlow	Appearance and motion	Two-step	online	disjointed
ELECTRICITY [44]	probabilistic	CityFlow	Appearance and motion	Two-step	online	disjointed
TrafficBrain [45]	clustering	CityFlow	Appearance and motion	Two-step	online	disjointed
MTMC_CDSC [46]	Quadratic program	DukeMTMC, MARS	Appearance and motion	Two-step	online	Disjointed
BIPCC [5]	Binary integer program	DukeMTMC	Appearance and motion	Two-step	online	Disjointed
SCLM + GAE [47]	Hierarchical Clustering	CityFlow	Appearance and topological	Two-step	online	Disjointed
TAREIDMTMC [15]	probabilistic	DukeMTMC-Re-ID	Appearance and motion	Two-step	online	Disjointed
OPA [48]	Maximum weighted bipartite matching	DukeMTMC, EPFL, CAMPUS, MOT15, MOT16, and MOT17	Appearance and motion	Two-step	online	Overlapping disjointed
mcmt [49]		CityFlow	Appearance	Two-step	online	Disjointed
TRACTA + SAP [50]	Clustering	CityFlow	Appearance and motion	Two-step	online	Disjointed
Fivefive [51]	–	CityFlow	Appearance	Two-step	online	Disjointed
Matcher [43]	Greedy search	CityFlow	Appearance and motion	Two-step	online	Disjointed
BOE [52]	Clustering	CityFlow	Appearance	Two-step	online	Disjointed
FraunhoferiOSB [53]	Clustering	CityFlow	Appearance and motion	Two-step	online	Disjointed

projected camera views. Then the pose estimation algorithm is used to detect key points in the human body, and a motion analysis algorithm is employed to track these points over time. The result is a sequence of human poses and movements that were represented in the scene-centered reference frame. First, clusters were created based on the extracted poses, and then associating these clusters across different camera views based on the spatio-temporal relationships between the poses, giving cross-camera trajectory for each person. The methods discussed so far used a global description to create tracklets, but this approach failed to capture the local similarity between targets, making the methods vulnerable to occlusion and fast motion. To address this issue, You et al. [48] proposed an online Optical-based Pose Association (OPA) technique for multi-target multi-camera tracking. OPA used local pose matching to tackle the occlusion problem and optical flow to reduce the distance caused by fast motion. OPA employed OpenPose to generate a human pose for each proposal and PWC-Net to produce an optical flow for adjusting the estimated pose from the previous frame. The modified Object Keypoint Similarity method computed the similarity between the pose of the current frame and the adjusted pose from the prior frame. The optical-based pose similarity was combined with the visual and bounding box spatial similarities to create the final similarity matrix, which was used in the Kuhn-Munkras algorithm for data association. However, the model showed poor performance for a crowded scenario that demanded the robust detection or data association method. After multiple attempts in research to enforce limitations like sparsity and time conflict, an elegant solution for implementing the principle of matching consistency is yet to be discovered.

The methods discussed so far follow the tracklet-tracklet matching approach. These methods faced two major problems: 1. As targets were observed in varying numbers of cameras, the local tracklets corresponding to each target were uncertain and differed in quantity. As a result, determining the appropriate number of tracklets to merge and form a global trajectory became a challenging task. 2. It was difficult to impose such a matching consistency principle systematically (e.g., if certain tracklets were grouped together to form a global trajectory for a specific target, then different sets of grouped tracklets should be mutually exclusive). To address the above problems, He et al. [35] proposed a novel

approach that formulates the cross-camera tracklet matching problem as a Tracklet-to-Target Assignment (TRACTA) problem, where each tracklet was allocated to a distinct target. They utilized the Restricted Non-negative Matrix Factorization (RNMF) algorithm to determine the optimal assignment. Despite achieving promising results, TRACTA was incapable of correctly identifying targets that had a similar appearance and were located close to each other. Therefore, it was necessary to incorporate more distinct features that can help distinguish between these targets. Already GNN proved their significance in SC-MPT to leverage the benefits of GNN into MC-MPT and to improvise the data association. Quach et al. [59] tackled the challenge of data association by framing it as a link prediction task on a graph. In their approach, the graph's nodes corresponded to individual tracks, and a predictor was created using a novel dynamic graph formulation. This formulation incorporated temporal data of an object over a period of time, as well as its connections to other objects, in order to improve identification assignment. By utilizing the feature representations and moving patterns of each object, their method outperformed existing state-of-the-art techniques. Although better-discriminating features were obtained with the attention model, the complexity of the model also increased. Following the GNN with a less complex model providing robust features would be appreciable for the real-time task.

The current available MC-MPT approaches had a high level of computational complexity and were not adequately robust in the previously mentioned challenges. In [60], a real-time Attribute Recognition-based MC-MPT (AR- MC-MPT) framework was proposed that aimed to overcome these limitations. This framework employed an end-to-end approach for object detection, feature extraction, and attribute recognition. By applying attributes, the online tracking performance of MC-MPT was significantly improved in the aforementioned challenges. The AR- MC-MPT pipeline comprised three modules. The first module employed a novel one-shot Single-Camera Tracking (SCT) architecture named Attribute Recognition-Multi Object Tracking (AR-MPT), which performed object detection, Re-ID feature extraction, and attribute recognition using one backbone via multi-task learning. To handle instances of one identity detected in overlapping areas of cameras, hierarchical clustering was performed in the second module. Finally, the third module employed a new data association algorithm using spatial information to reduce matching candidates. Additionally, an efficient strategy was proposed in the data association algorithm to remove lost tracks by striking a balance between the number of lost tracks and the maximum lost time. The proposed model significantly reduced complexity and achieved a 20% improvement in IDF1 score over the existing methods.

Despite having several successful solutions for MC-MPT, one of the key disadvantages of the existing technique was that there was often a trade-off between accuracy and computational efficiency in existing methods. Some methods sacrificed accuracy for faster processing time, while others prioritized accuracy at the cost of computational efficiency. Since object tracking was a real-time application balancing the trade-off between accuracy and complexity was much essential. Also, there was still room for further improvement in tracking accuracy in a cross-camera network. In Table 1, we provide a comprehensive list of all reviewed trackers, and a detailed comparison of each tracker on six different factors are summarized.

III. METHODOLOGY

As shown in Figure 1, we follow two stage MC-MOT approach. In the first stage, the local tracklets are generated. In a multi-camera network, each person is under the coverage of C cameras, represented by the set $C = c_1, \ldots, C_n$. The local tracklets are generated from each camera. We use the DeepSort tracker with YOLOX as the detector and adopt the CNN architecture proposed in [61] for feature extraction. Both the detector and feature extractors are supervised on multi-source datasets. At time step t, the multi-object tracking is enforced to generate the set of local tracklets, i.e, $L_c^t = l_j^t$. Then the dynamic graph is initialized with the feature embeddings of each tracklets at time t. Further, in the second stage for cross-camera tracklet association, our method embeds a set of local tracklets into a dynamic message-passing neural network, which generates richer tracklet representations and computes the similarity simultaneously between the set of unassigned local tracklets l_j^t and set of known tracklets $U_{c=1}^{C} U_{k=1}^{t-1} L_c^{(k)}$ obtained from previous time steps. Followed by the similarity of two local tracklets is given as input to the classifier, which provides the best link prediction accuracy. A detailed description of each module is provided in the subsections.

A. LOCAL TRACKLET GENERATION

We use the DeepSort [62] for single-camera tracklet generation. We provide a concise overview of DeepSORT, presenting it as a dual-branch framework comprising an appearance branch and a motion branch. The performance of the tracking highly depends on the detector used to localize the object in a multi-camera network. We use YOLOX for person detection to generate optimal object detection. The detector is trained on four public benchmarks to make it more generalizable. We employ the CNN architecture with low computational complexity recommended in [61] for the extraction of appearance feature and the feature extractor is fine tuned on multi-source datasets. In the appearance branch, for each frames detections, a deep appearance descriptor (a simple Convolutional Neural Network) pretrained on the person re-identification dataset is employed to extract appearance features. This process utilizes a feature bank mechanism to store the features from the last 100 frames for each tracklet. As new detections emerge, the smallest cosine distance between the feature bank Bi of the i^{th} tracklet and the feature f_j of the j^{th} detection is calculated. The distance serves

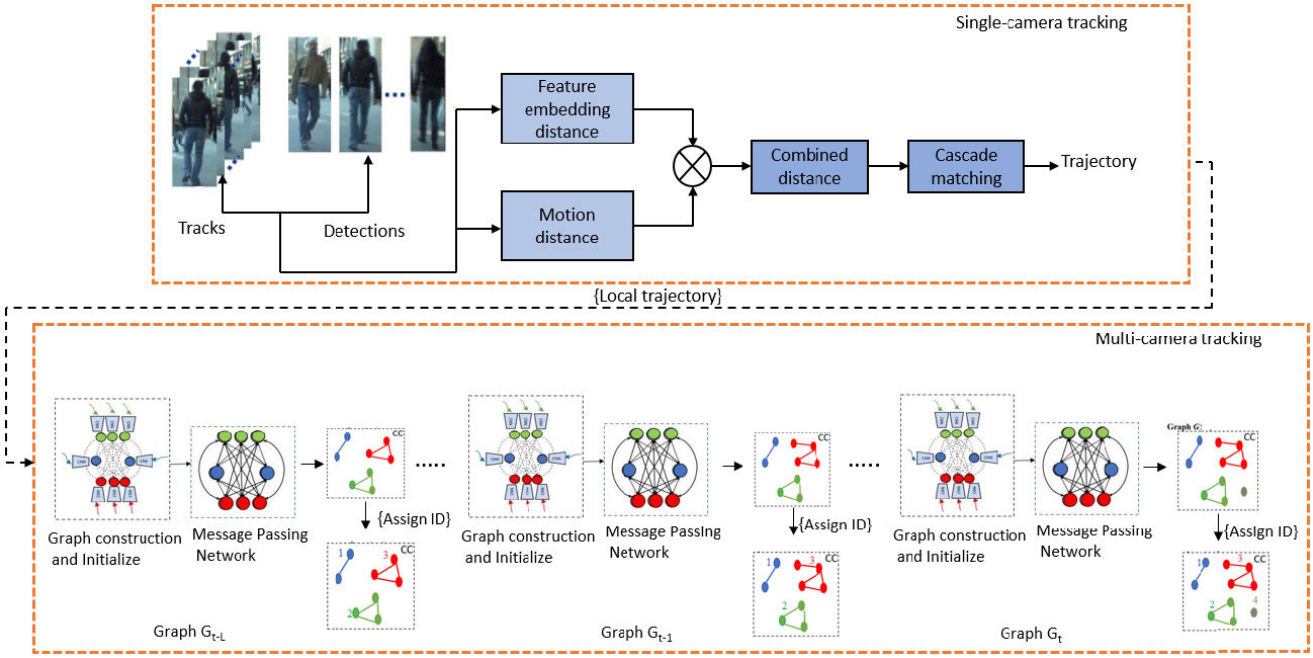

FIGURE 1. Proposed framework for multi-camera multi-object tracking.

as the matching cost throughout the association process as shown in Equation 7.

$$d\,(i,j) = min\left(1 - f_j^T f_k^{(i)})|f_k^{(i)} \epsilon B_i\right) \tag{1}$$

In the motion branch, the Kalman filter algorithm [24] is employed to predict the positions of tracklets in the current frame. This process involves a twophase approach, comprising state prediction and state update. During the state prediction step, it forecasts the current state as:

$$\hat{x}_k' = F_k \hat{x}_{k-1} \tag{2}$$

$$\hat{P}_k = F_k P_{k-1} F_k^T + Q_k \tag{3}$$

Here, $\hat{x}_{k-1}$ and P_{k-1} represent the mean and covariance of the state at time step $k-1$, while $x\hat{}k$ and P_k denote the estimated state at time step k. The state transition model is represented by F_k, and Q_k stands for the covariance of the process noise. In the state update step, the Kalman gain is computed based on the covariance of the estimated state P_k and the observation noise R_k as:

$$K = P_k' H_k^T \left(H_k P_k' H_k^T + R_k\right)^{-1} \tag{4}$$

Here, H_k^T represents the observation model, which transforms the state from estimation space to observation space. Subsequently, the Kalman gain K is employed to update the final state:

$$x_k = \hat{x}_k' + K\left(z_k - H_k \hat{x}_k'\right) \tag{5}$$

$$P_k = (I - KH_k)\, P_k' \tag{6}$$

B. DYNAMIC MESSAGE PASSING NEURAL NETWORK

In order to predict and assign IDs to new nodes in graph G_t at time step t, we learn dynamic graph representation with two-time steps $\left\{G^{(t-2)}, G^{(t-1)}\right\}$. Therefore, we conduct MC-MOT experiments by learning dynamic graph representation with time step L=3. As shown in Figure 1, a graph G(t) = (V(t); E(t)) (V is vertices, E is edges) is built at a specific time step t, with the vertex set V_t Containing all the tracklets tracked up to that point. The number of vertices in our graph is expanding over time because new vertices are being added at each time step t, as shown by the equation V(t) = V(t−1) $\bigcup$ N(t), where N(t) stands for the set of new vertices. The graph estimation at time step t follows three steps such as (1) graph construction and initialization, (2) message passing, and (3) classification. In the first step, the graph is constructed by initializing each node with its feature vector obtained by CNN. Then in the message-passing step, each node exchanges a message (feature) with its neighborhood for L iteration. The feature embedding obtained at the last iteration is passed to the classifier for prediction.

1) DYNAMIC GRAPH CONSTRUCTION AND INITIALIZATION

Each node (tracklets) is initialized by the feature vector h_{v_i}, that is obtained by passing nodes features through a learnable encoder E_v as defined in Equation 7:

$$h_{v_i} = E_v(CNN(v_i)) \tag{7}$$

The initial edge embeddings are created by combining the visual and spatial characteristics of the two nodes that the edge connects. Equation 8 specifies the appearance similarity

computation between two nodes.

$$\Delta f_{i,j} = Cosine(CNN(v_i), CNN(v_j)) \tag{8}$$

It is necessary to transform detection coordinates to a common ground plane to determine the relative spatial distance between detections from various cameras. Each node will have its detection coordinates (x_i, y_i, w, h) that are projected to the ground plane using the homography matrix of a camera as defined in Equation 9.

$$(X_i, Y_i) = H_{c_i}\,(x_i + w_i/2, y_i) \tag{9}$$

where, H_{c_i} is the homography matrix of camera c_i. In accordance with Equation 10, we determine the relative spatial similarity between two nodes.

$$\Delta S_{i,j} = \left\| (X_i, Y_i)\,, (X_j, Y_j) \right\| \tag{10}$$

In order to initialize the edge, the concatenation of $\Delta f_{i,j}$ and $\Delta S_{i,j}$ is sent to a learnable edge encoder E_e as defined in equation 11 that further updates the edge embeddings during the message passing step.

$$h_{(v_i,v_j)} = E_e\left(\Delta f_{i,j}, \Delta S_{i,j}\right) \tag{11}$$

2) MESSAGE PASSING NETWORK

Once the graph is constructed and initialized with initial state embeddings, MPN propagates the neural information with adjacent nodes and edges over the graph G_t at time t. This message exchange helps to update the nodes and edge state. Every node and edge compute the messages it has delivered and received at each propagation step, then aggregates the messages it has received, and lastly, update its representation by fusing the new data with the old. Node update and edge update are the two stages of the message passing step, and both updates are performed across L iterations. The edge connecting two nodes (v_i, v_j) and its embeddings are updated as defined by Equation 12 for each iteration of L ϵ [1, L]:

$$h^l_{(v_i,v_j)} = \upsilon_e\left[h^{l-1}_{v_i}, h^{l-1}_{v_j}, h^{l-1}_{(v_i,v_j)}\right] \tag{12}$$

where, υ_e learnable edge encoder(FC+ReLU).

Each node's embeddings are updated by aggregating the incoming messages as defined by Equation 13.

$$h^l_{v_i} = \sum_{j \epsilon N(v_i)} m^l_{(v_i,v_j)} \tag{13}$$

where N is neighbor nodes and

$$m^l_{(v_i,v_j)} = \upsilon_v\left[h^{l-1}_{v_i}, h^l_{(v_i,v_j)}\right]$$

U_e and U_v are the edge, node update functions exchange the message across G_t for its state update. The message exchange for nodes and edges happens simultaneously in G_t. The learnable encoders used by the network and classifier are considered from [63].

C. LINK PREDICTION

Once the nodes and edges update their states for the L iteration, the feature embedding from the last iteration is used for prediction. For each ground truth label $y_{(v_i,v_j)}$, the model is trained to predict the probability $\hat{y}_{(v_i,v_j)}$ of an edge belongs to the same identity. With $y_{(v_i,v_j)}$, serving as labels, it can be viewed as an edge classification task. According to each edges prediction, the graph is trimmed during inference. At iteration l, the classification of a given edge is calculated as defined in Equation 14:

$$\hat{y}_{(v_i,v_j)} = C\left(h^l_{(v_i,v_j)}\right) \tag{14}$$

where C is the classifier (FC + ReLU) followed by the sigmoid function yields a single prediction value.

1) TRAINING

Training loss of a graph G_t at time t is computed using cross-entropy (CE) loss function for all edges and iterations as defined in Equation 15. By doing edge classification, we ultimately learn a strategy that can directly predict graph partitions.

$$L_{Gt} = \sum_{l=1}^{L} \sum_{(v_i,v_j)\epsilon E} CE\left(\hat{y}_{(v_i,v_j)}, y_{(v_i,v_j)}\right) \tag{15}$$

2) INFERENCE

Our objective is to collect as many connected components (CCs) as possible by grouping the nodes in the graph G_t that correspond to the similar person identification at time t. First, the local tracklets are generated to infer these graph components that will form the vertices, and the connections between them are edges. The messages are exchanged with its N neighborhoods for L iterations. We take into account the MPN model's output at the last iteration for edge prediction. As a result, we can calculate the likelihood that each edge in a graph G_t will be active or not active using the formula $\hat{y^L_{(v_i,v_j)}} \epsilon$ [0, 1]. The final prediction is binarized to classify whether the edge is active or non-active.

$$\hat{y^B_{(v_i,v_j)}} = \begin{cases} 0 & \hat{y^L_{(v_i,v_j)}} < 0.5 \\ 1 & otherwise \end{cases}$$

$\hat{y^B_{(v_i,v_j)}}$ It signifies whether the edge between the two nodes is active or inactive. While the active edges are preserved, the inactive edges are trimmed.

Post-processing is the last phase of inference. We operate on the presumption that each node can link to additional M-1 nodes, where M is the total number of views. The node that does not comply with this condition and if there exists a bridge, by removing one bridge having minimum predicted probability, the condition can be satisfied. If there is no bridge in the graph, an edge with a minimum predicted probability could be removed to hold the condition.

TABLE 2. Dataset split for training and Inference for EPFL and CAMPUS datasets, in each dataset setting (d1, d2, d3, d4), the green tick indicates the inference set, and the black color tick mark represents the train set.

Dataset	EPFL				CAMPUS			
	Laboratory	Terrace	Basketball	Passage	Garden1	Garden2	Parkinglot	Auditorium
d1	✓	✓	✓	✓	✓	✓	✓	✓
d2	✓	✓	✓	✓	✓	✓	✓	✓
d3	✓	✓	✓	✓	✓	✓	✓	✓
d4	✓	✓	✓	✓	✓	✓	✓	✓

D. CONNECTED COMPONENT

We estimate the CCs over the graph at each time step t after the classifier identifies clusters of nodes with multiple views of the same individual. A graph's component, or split, is referred to as a CC if there is a path connecting every pair of nodes. As a result, we obtain a list of connected components at each time step t. Thus, each component will be assigned a global identity number.

IV. RESULTS AND DISCUSSION

In this section, we conduct an experiment to demonstrate the benefits of training the detection and feature extractor on multiple datasets. Also, the effectiveness of the dynamic graph model with MPN to perform cross-camera association and prediction is examined. The performance of MC-MOT tracking is evaluated using CLEAR MOT metrics that evaluate for MOTP, MOTA, ID switch, IDF1, precision, and recall. The Multi-Camera Tracking Accuracy (MCTA) is compared with prior arts for overlapping and non-overlapping scenarios.

A. DATASETS AND IMPLEMENTATION DETAILS

1) DETECTOR

The detector is YOLOX with YOLOX-X as the backbone and COCO-pretrained model as the initialized weights. For pedestrian detection, we use a YOLOX detector. To increase the detector's generalizability, we train it on a huge dataset (109,471 total images) made up of four publicly available pedestrian benchmarks: the CrowdHuman [64], the CUHK-SYSU [65], the PRW [66], and the Caltech dataset [67]. The detector is trained by setting the detection threshold to 0.5. The input image size is 1440 $\times$ 800and the shortest side ranges from 576 to 1024 during multi-scale training. The data augmentation includes Mosaic and Mixup. The model is trained on 8 NVIDIA Tesla V100 GPU with batch size of 48. The optimizer is SGD with weight decay of 5×10^4 and momentum of 0.9. The initial learning rate is 10^3 with 1 epoch warmup and cosine annealing schedule and trained the detector for 50 epochs.

2) FEATURE EXTRACTOR

The CNN architecture recommended in [61] is employed for extracting appearance features, and it undergoes training using multi-source datasets such as Market1501, CUHCK03, and the MARS person re-identification dataset, encompassing 1,117,655 bounding boxes corresponding to 4,229 pedestrians. Local tracklets, representing ReID feature vectors, are obtained at each timestep using the DeepSort Tracker. These 512-dimensional tracklets initialize the graph. On top of OSNet, a classification layer (linear fully connected + softmax) is added. Person matching relies on cosine distance utilizing 512-dimensional feature vectors from the last fully connected layer. The batch size and weight decay are set to 64 and 10^4, respectively, with images resized to 256 $\times$ 128. For training from scratch, SGD is employed, optimizing the network for 350 epochs. The learning rate begins at 0.065 and decreases by 0.1 at the 150^{th}, 225^{th}, and 300^{th} epochs. Data augmentation involves random flip, random crop, and random patch. For fine-tuning, the network is trained using AMSGrad with an initial learning rate of 0.0015 for 250 epochs. The learning rate undergoes decay using the cosine annealing strategy, without restarting. During the initial 10 epochs, the ImageNet pre-trained base network is frozen, allowing only the randomly initialized classifier to be open for training [68]. Data augmentation in this phase includes random flip and random erasing [69].

3) DYNAMIC MESSAGE PASSING GRAPH NEURAL NETWORK

The GNN is trained to learn features and similarity measures simultaneously. We use four benchmarks, CAMPUS [16], EPFL [39], MCT [6], and PETS09 [1], to evaluate the MCT performance for overlapping and non-overlapping scenarios. The scenes used for training and inference are shown in Table 1 and Table 2 for EPFL, CAMPUS, and MCT datasets, respectively. To evaluate the performance of the PETS09 dataset, we have utilized the model-trained CAMPUS. In MC-MOT, by learning a dynamic graph representation with two-time steps, $G(t-2), G(t-1)$, we can anticipate and assign IDs to new nodes in Gt at time step t. Thus, to learn every parameter of the learnable encoders and the classifier, training data are divided into mini-batch with a chunk size of 3, and a mini-batch gradient descent using the Adam optimizer is used. The proposed method is implemented in the PyTorch on NVIDIA GForce GTX, 1080. The training process involves a maximum of 100 epochs, utilizing a batch size of 512 chunks. It's important to note that padding is applied to amalgamate these chunks, accommodating varying numbers of nodes, into a cohesive batch for training. Subsequently, we select the top-performing model based on validation set results for evaluation across four benchmark datasets. The scenes used for training and inference are shown in Table 2 and Table 3 for EPFL, CAMPUS, and

TABLE 3. Dataset split for training and Inference for MCT datasets, in each dataset setting (d1, d2, d3, d4), the green tick indicates the inference set, and the black color tick mark represent the train set.

Dataset	MCT			
	Dataset1	Dataset2	Dataset3	Dataset4
d1	✓	✓	✓	✓
d2	✓	✓	✓	✓
d3	✓	✓	✓	✓
d4	✓	✓	✓	✓

MCT datasets, respectively. To evaluate the performance of the PETS09 dataset, we have utilized the model-trained CAMPUS dataset.

B. ABLATION STUDY

In this section, we analyze and justify the competitive performance that our method achieved. This section primarily seeks to illustrate the following desirable qualities of the proposed technique. (1) Robust feature representations. (2) Consistent object detection and tracking (3) Lightweight tracker.

1) ROBUST FEATURE REPRESENTATIONS

It is a fact that the accuracy of data association is highly influenced by the model considered for feature representation. To improve feature extraction ability, we train the detector and feature extractor on multi-source datasets. Then, conduct a comparative study on the features extracted by the model trained on a single source vs. multiple sources. As shown in Figure 2, the features extracted by the models are utilized to generate a heatmap. The heatmaps in the first row (Figure 2 (a)) are generated using the model trained on a single source. We find background influence in feature learning (as highlighted in yellow). In contrast, second-row (Figure 2 (b)), heatmaps are generated using the model trained on multiple sources. Figure 2 illustrates that the features extracted by the model trained on multiple sources are better clustered than the one trained on a single source. Also, the background feature influence is reduced when the model is trained with multiple sources.

Further, the better feature representation ability of the proposed approach is analyzed by plotting t-SNE. The representations of all the nodes over the entire video sequence produced by our method and the original node features obtained by the feature extractor are plotted in Figure 4 (a) and Figure 4 (b), respectively, while Figure 4 (c) represents the features obtained by the detector. The t-SNE plot shown in Figure4 (a) signifies that MPGNN-generated features are better clustered than the original. Similarly, the original features extracted by the ReID model trained on a multi-source dataset are better cluttered than the detector-generated features.

Also, the impact of using GNN to generate a rich representation for cross-camera tracking is analyzed with qualitative track results shown in Figure 3. The first image

TABLE 4. MOT metric comparison with and without the inclusion of dynamic GNN evaluated on Auditorium sequence from CAMPUS dataset. IDF1- identification F1 score, IDP- identification precision, IDR- identification recall, IDs-ID switches.

Features	IDF1 ↑	IDP ↑	IDR ↑	IDs ↓
Feature Extractor (CNN)	49.5	50.1	49.0	135
DGNN	**63.2**	**70.5**	**56.2**	**46**

TABLE 5. Track quality measures on EPFL dataset. IDf1-identification F1 score, MT- mostly tracked, ML-Mostly Lost, IDs-ID switches.

Sequence	EPFL				
	IDF1 ↑	MT ↑	ML ↓	PT ↓	IDs ↓
Terrace	38.4	8	0	0	27
Basketball	28.8	13	0	2	42
Laboratory	32.0	1	0	5	18
Passageway	29.9	4	0	2	18

is the tracking results obtained using the feature extractor, excluding the learnable encoders. In contrast, the second-row image is the tracking results with the inclusion of a dynamic GNN. In frame 2487, we find more persons missed from tracking (highlighted with a red dotted box), whereas the proposed model notably covers the number of missed detections (frame 2487 second row). Further, we quantitatively justify the role of GNN in cross-camera tracking by the results shown in Table 4. The results obtained with DMGNN are evident that the inclusion of a dynamic GNN has been helpful in extracting better features for discrimination than barely using the features from simple CNN.

2) CONSISTENT OBJECT DETECTION AND TRACKING

The detection quality is analyzed by visualizing the detection results of YOLOX by training it on a crowd-human dataset. Also, to evident the improved generalization capability of the detector, the qualitative results are analyzed. As shown in Figure 5, the variation is found in the pedestrian detection results when the YOLOX is trained only on crowd humans and on multiple datasets. The qualitative results are shown in Figure 5 for the auditorium sequence from CAMPUS and the basketball sequence from the EPFL dataset. Figure 5 (a) illustrates the detection results obtained from a model trained on one dataset, whereas Figure 5 (b) represents the detection results of the generalized model. The bounding box in red represents missed detections. The respective person in the same frame has been detected by the generalized detector. These qualitative results reveal that the detector trained on a multi-source is more beneficial than training on a single source. The tracking performance directly depends on the quality of detections provided for the data association stage. Therefore, the generalized detector will contribute to enhancing the tracking performance.

Additionally, the detector's performance is assessed by quantitatively analyzing the results obtained from EPFL

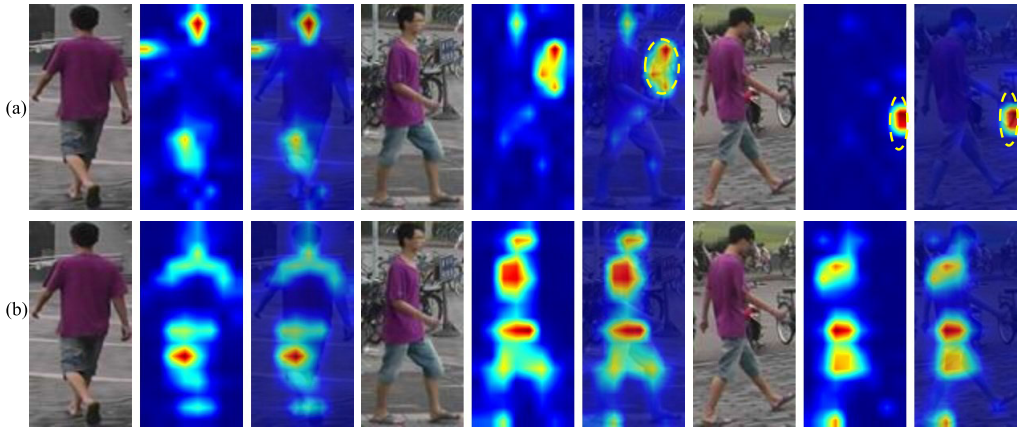

FIGURE 2. Heatmap visualization (left to right- original image, heatmap, overlapped image): (a) Generated using single source model (b) Generated using multi-sourced model.

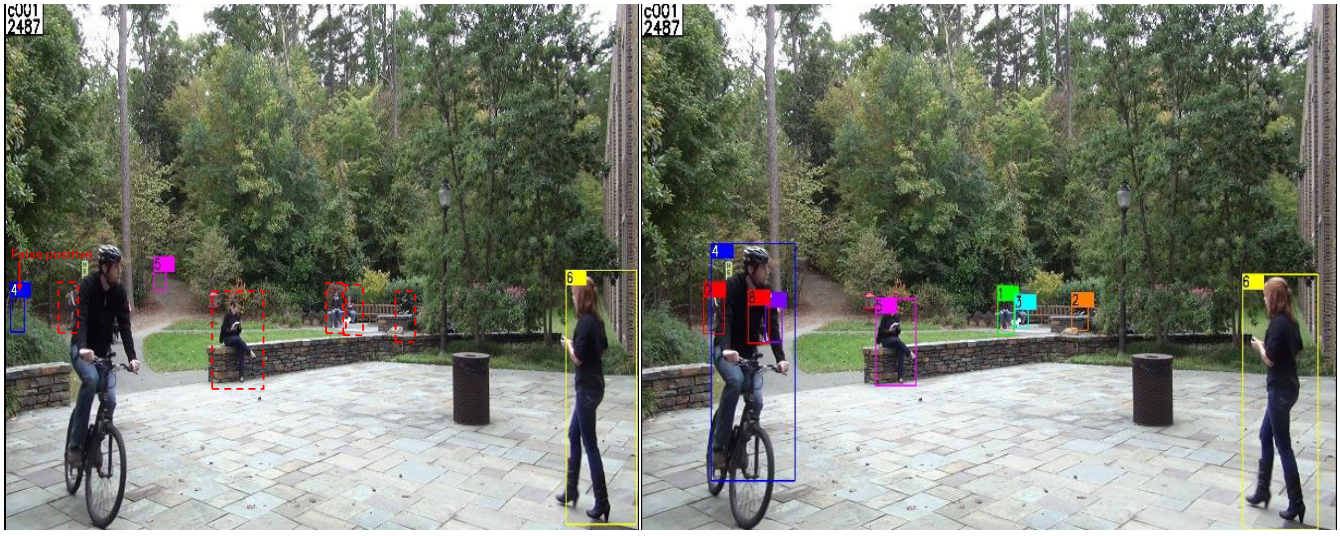

FIGURE 3. Qualitative track results: (a) tracking using baseline model (excluding GNN) model (b) tracking using dynamic GNN model.

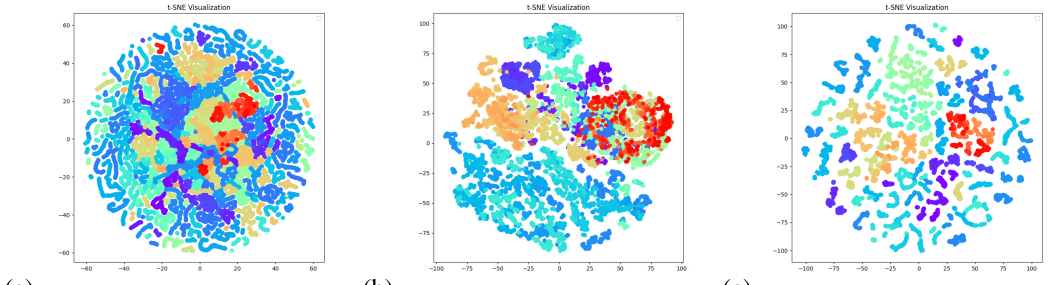

FIGURE 4. Terrace video sequence: (a) MPGNN features (b) ReID model features (c) YOLOX features.

dataset. This evaluation is conducted using both single and multi-source supervised models, and the corresponding findings are presented in Table 6. The result obtained using multi-sourced model reveals the improved generalization ability of the detector.

To show the consistent tracking achieved by the proposed method, the qualitative results of a laboratory and terrace video sequences are shown in Figure 6, Figure 7. A square with four grids represents the same person tracked in four different cameras. Each person is localized, and

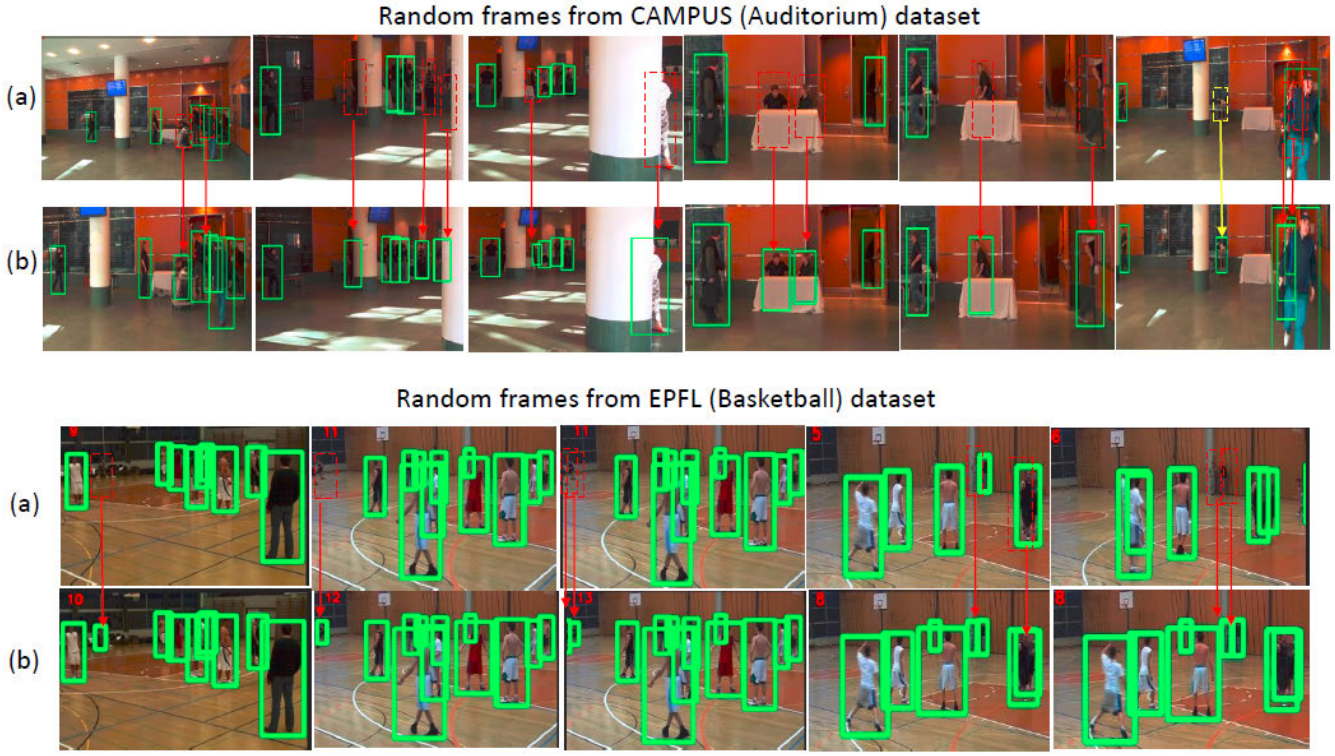

FIGURE 5. Qualitative object detection results on Auditorium and Basketball sequence (a) YOLOX trained on a single dataset. (b) YOLOX trained on multiple datasets.

TABLE 6. The detector's performance on EPFL dataset using both single and multi-source supervised models. All the results are in [%].

Approach	Precision	Recall	mAP
Single source	86.0	83.2	79.6
Multi-source	86.5	90.1	89.9

TABLE 7. Track quality measures on CAMPUS dataset. IDf1-identification F1 score, MT- mostly tracked, ML-Mostly Lost, IDs-ID switches.

Sequence	CAMPUS				
	IDF1 ↑	MT ↑	ML ↓	PT ↓	IDs ↓
Garden1	29.6	13	0	3	46
Garden2	30.6	17	0	1	53
Auditorium	63.2	21	0	0	46
Parkinglot	30.2	13	0	2	43

unique identification (ID) number is assigned. The tracker achieved consistency in tracking by assigning the same person ID. Our tracker is efficient in tracking during partial occlusion scenarios (frame 282, c003, ID 0) and marginal overlapping of multiple persons (frame 2020 in all cameras). However, our tracker also has some shortcomings, such as some false negatives detected (on frame 1300, c002, ID 2) due to over-supervising the model and IDs during tracking.

The quantitative results of CAMPUS and EPFL datasets are shown in Table 5 and Table 7. The proposed tracker proved its significance by improving the track quality (MT, ML, PT) and less the number of IDs. The evaluation covers multiple matrices that help the research community conduct a comparative study.

Furthermore, our proposed method yields SCT results on EPFL, CAMPUS, and PETS datasets, as presented in Table 8. The quantitative outcomes for single-camera MOT demonstrate the robustness of our approach in tracking individuals within a single camera. The inclusion of a generalized detector and ReID model has contributed to these promising results. Moreover, the implementation of a MPGNN further augments the feature representation ability. As a result, our approach attains higher MOTA and IDF1 scores, accompanied by commendable track quality metrics (MT, ML, PT) and zero instances of IDs.

3) LIGHTWEIGHT TRACKER

The tracker is built following tracking by detection paradigm, in which we have considered a single-stage detector, i.e., YOLOX, which is computationally less complex and faster than two-stage detectors. The current best performing MC-MOT tracker is DyGLIP [59] that has considered Mask-RCNN for detection. In comparison to Mask-RCNN YOLOX is much faster and anchorless. Also, we have considered the CNN architecture proposed by [61] for feature extractions, which takes 1.9 million parameters. Whereas, the CNN considered in DyGLIP [59] is trained on imagenet dataset and it takes 2.2 million parameters.

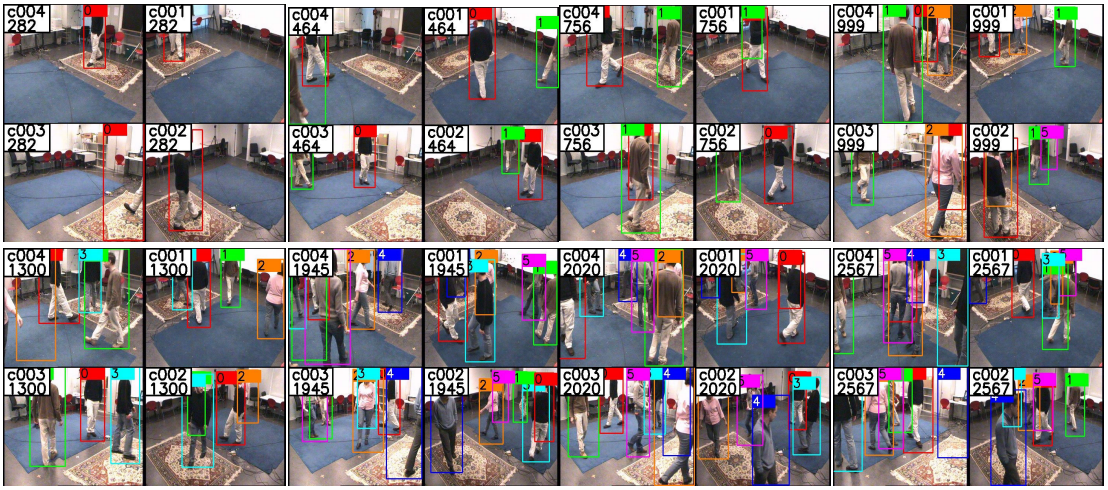

FIGURE 6. Qualitative results of Laboratory video sequence (random frames are considered for visualization).

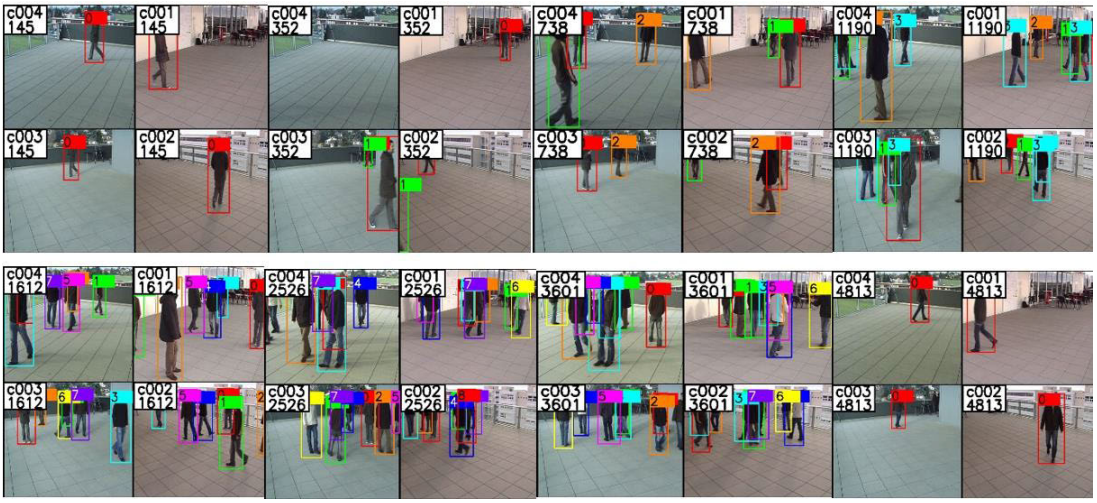

FIGURE 7. Qualitative results of Terrace video sequence (random frames are considered for visualization).

TABLE 8. Performance of the proposed method on single-camera MOT on EPFL, CAMPUS, and PETS.

Dataset	Sequence	IDF1 [%] ↑	MT↑	ML↓	PT↓	IDS↓	MOTA [%]↑
EPFL	Basketball	94.9	51	2	4	0	90.3
	Laboratory	78.4	10	4	10	0	64.5
	Passageway	93.6	19	1	4	0	87.9
	Terrace	95.1	30	0	6	0	90.6
CAMPUS	Auditorium	99.6	67	0	0	0	99.2
	Garden1	93.6	50	1	11	0	87.9
	Garden2	96.7	62	0	9	0	93.6
	Parkinglot	95.6	51	1	6	0	91.5
PETS09	S2L1	92.9	39	0	11	0	86.8

In addition, our proposed dynamic graph neural network has only 0.2M parameters, making the model light and faster in training.

C. COMPARISON WITH PRIOR ARTS

The performance of our approach is evaluated on CAMPUS, EPFL, and PETS09 datasets containing overlapping fields of view (FOVs). We also check the tracking performance on the MCT dataset collected in non-overlapping FOVs. We contrast our method with other MC-MOT approaches, such as KShortest Path (KSP) [38], Hierarchical Composition of Tracklet (HCT) [16], Spatio-Temporal Parsing (STP) [58], and Branch-and-Price (B&P) [18], TRACTA [35], which tracks for overlapping FOVs between various camera

TABLE 9. Comparison of evaluation results on EPFL dataset with the Prior-Art techniques (all the results are in %).

Seq	Method	MOTA ↑	MOTP ↑
Passageway	KSP [38]	40	57
	HCT [16]	44	71
	TRACTA [35]	52.1	77.5
	OPA [48]	67.3	80.3
	DyGLIP [59]	70.4	97.2
	Li [70]	83.4	78.5
	Gan [71]	77.0	74.0
	Proposed	**90.3**	**99.1**
Basketball	KSP [38]	56	54
	HCT [16]	60	68
	TRACTA [35]	64.3	72.5
	OPA [48]	61.9	74.8
	DyGLIP [59]	66.3	89.5
	Li [70]	79.1	72.7
	Gan [71]	78.4	72.6
	Proposed	**87.9**	**98.3**

TABLE 10. Comparison of performance on PETS09 dataset with the Prior-Art techniques (all the results are in %).

Sequence	Method	MOTA↑	MOTP ↑
S2L1	KSP [38]	80	57
	B&P [18]	72	53
	HCT [16]	89	73
	TRACTA [35]	87.5	79.2
	LMGN [72]	70.4	73.2
	OPA [72]	88.6	81.1
	DyGLIP [59]	93.5	94.7
	Li [70]	86.2	79.1
	Proposed	88.7	**99.8**

views. The MC-MOT evaluation results of EPFL, PETS09, and CAMPUS are summarized in Tables 9, 10, and 11, respectively. Our proposed method was compared with state-of-the-art methods using the evaluation metrics that were available. As a result, Tables 9, 10, 11, and 12 contain a variety of evaluation metrics for comparison.

Most of the existing studies evaluated their performance on highly challenging video sequences such as Basketball and passageway. These sequences are heavily crowded, and the visibility of a person is not much clear in the passageway dataset. As shown in Table 9, our approach improved MOTA by 8.9% and 21.6% on passageway and basketball sequences over the second-best methods. The HCT and OPA methods rely on associating individuals using similarities in appearance and pose, which are derived from their personal characteristics. However, these approaches exhibit limited performance in crowded scenes due to the poor distinction in location differences. Additionally, they struggle with insufficient lighting conditions and uniform clothing. Furthermore, these methods require a longer processing time due to the extraction of multiple features. In contrast, our method exploits the target correspondence by exchanging messages between each node. This approach proves to be robust even in scenarios where individuals are captured under

TABLE 11. Comparison of evaluation results of the CAMPUS dataset with the Prior-Art techniques (all the results are in %).

Sequence	Method	MOTA ↑	MOTP ↑	MT ↑	ML ↓
Garden 1	HCT [16]	49	71.9	31.3	6.3
	STP [58]	57	75		–
	TRACTA [35]	58.5	74.3	30.6	1.6
	OPA [72]	72.8	78.5	-	-
	DyGLIP [59]	71.2	91.6	31.3	0.0
	Li [70]	80.4	73.3	31.3	0.0
	Gan [71]	80.1	72.6	-	-
	Proposed	**87.9**	**99.6**	**80.6**	**1.6**
Garden 2	HCT [16]	25.8	71.6	33.3	11.1
	STP [58]	30	75	–	–
	TRACTA [35]	35.5	75.3	16.9	11.3
	OPA [72]	53.2	77.4	-	-
	DyGLIP [59]	87.0	98.4	66.67	0.0
	Li [70]	76.7	76.4	-	-
	Gan [71]	59.9	70.2	-	-
	Proposed	**93.6**	**99.9**	**94.4**	**0.0**
Auditorium	HCT [16]	20.6	69.2	33.3	11.1
	STP [58]	24	72	–	–
	TRACTA [35]	33.7	73.1	37.3	20.9
	OPA [72]	24.3	77.9	-	-
	DyGLIP [59]	96.7	99.5	95.24	0.0
	Proposed	**99.2**	**100**	**100**	**0.0**
Parkinglot	HCT [16]	24.1	66.2	6.7	26.6
	STP [58]	28	68	–	–
	TRACTA [35]	39.4	74.9	15.5	10.3
	OPA [72]	55.6	78.1	-	-
	DyGLIP [59]	72.8	98.6	26.67	0.0
	Proposed	**91.5**	**99.9**	**86.6**	**0.0**

varied illumination and are located in close proximity to each other. Regarding cross-camera association, OPA adopts a research approach that involves generating target trajectories within each individual camera and then performing associations across multiple cameras as a whole. However, this method fails to utilize the complementary information present between cameras, leading to missed detections and identity switches. Consequently, OPA yields a lower MOTA value. Similarly, the performance of Gan et al. [71] falls short compared to our method. Gan et al. [71] extract visual features from each detected subject individually, which leads to instability in the features themselves. Consequently, these unstable features pose challenges for cross-camera associations. Overall, the comparative results demonstrate that the proposed method is superior to the existing techniques.

Whereas on the PETS09 dataset, the proposed tracker faced frequent IDs. Thereby, the model tracking accuracy is in second place among the prior arts. As listed findings in Table 10, the detection performance is still outperforming existing methods.

The tracker achieved remarkable progress in MOTA, precision (MOTP), and tracking quality (MT, ML) as demonstrated in Table 11. These video sequences are collected in challenging scenarios such as light variation, partially in similar cloth patterns, and occluded by vehicles. The proposed tracker achieved improved performance on all the metrics compared to existing techniques. Moreover, the majority of these comparative methods employ a two-stage approach, which involves feature extraction for

TABLE 12. Comparison of evaluation results of MCT dataset with the Prior-Art techniques.

Sequence	Method	MOTA ↑	MOTP ↑	Precision ↑	Recall ↑	IDS ↓
Dataset1	EGM [6]	59.4	68.0	79.7	59.2	1888
	RAC [73]	92.6	64.6	69.2	60.6	154
	ICLM [57]	87.3	68.1	77.2	60.9	112
	TRACTA [35]	94.9	85.2	92.7	92.6	
	DyGLIP [59]	86.7	97.0	93.4	86.8	37
	Proposed	**96.9**	**98.3**	**97.1**	**95.9**	**39**
Dataset2	EGM [6]	67.2	70.6	79.8	63.3	1985
	RAC [73]	86.8	73.7	69.5	78.4	171
	ICLM [57]	88.3	76.6	83.3	70.9	123
	TRACTA [35]	93.4	85.9	95.5	95.4	60
	DyGLIP [59]	95.7	96.8	97.7	95.9	101
	Proposed	**99.9**	**100**	**100**	**100**	**63**
Dataset3	EGM [6]	27.0	64.7	82.1	53.5	525
	RAC [73]	9.2	55.3	47.5	66.2	666
	ICLM [57]	53.2	69.1	66.0	72.6	228
	TRACTA [35]	58.5	75.4	75.2	91.3	144
	DyGLIP [59]	92.7	96.5	98.2	93.7	122
	Proposed	**94.8**	**98.1**	**100**	**100**	**36**
Dataset4	EGM [6]	35.8	71.1	83.6	61.9	3111
	RAC [73]	53.9	63.0	52.2	79.4	329
	ICLM [57]	62.5	86.8	87.6	86.0	189
	TRACTA [35]	79.6	90.0	86.3	96.0	70
	DyGLIP [59]	92.5	96.6	91.3	92.9	100
	Proposed	**96.9**	**99.4**	**97.9**	**98.3**	**56**

each estimated local trajectory and similarity measure for cross-camera trajectory. Whereas, our method extracts the features and similarity measures simultaneously without relying on unsupervised similarity metrics such as Euclidean, cosine, etc.

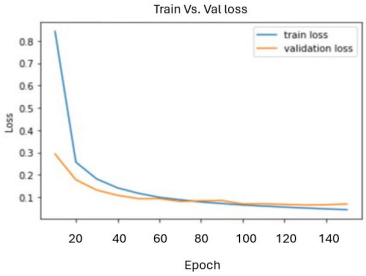

FIGURE 8. Train and Validation loss of MCT dataset.

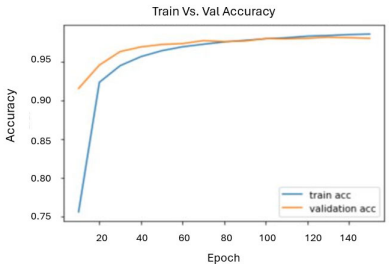

FIGURE 9. Train and Validation accuracy of MCT dataset.

The robustness of the proposed tracker is evaluated on the dataset collected under non-overlapping FOVs, and the MCT is one such dataset. The tracking results on the MCT dataset are reported in Table 12. The performance of the tracker is evaluated for MOTA, MOTP, detector precision, and recall. The results evidence the robustness of the proposed tracker over all the metrics compared to existing methods.

Our proposed model has achieved remarkable results, achieved 100% precision and recall on subsets Dataset2 and Dataset3 of the MCT dataset. These subsets predominantly have 2 to 3 persons per frame, with no challenging scenarios such as occlusion, motion pattern changes, or appearance variations. Therefor the detector has shown tremendous detection accuracy. Which in turn helped to achieve 99.9% of MOTA. However, other subsets exhibit higher density per frame (25 persons per frame), leading to intrinsic occlusion among multiple people. Additionally, there are partial appearance similarities among individuals in the other subsets. Despite these challenges, our model's performance remains strong, especially on Dataset2 and Dataset3. The less challenging nature of the video sequences in Dataset2 and Dataset3 contributes to the high precision and recall accuracy observed in these subsets. This emphasizes the importance of considering varying degrees of complexity in dataset subsets when evaluating tracking algorithms. The train-validation loss curves and train-validation accuracy curves are shown in Figure 8 and Figure 9, respectively. This signifies that there is no model overfitting on the dataset happened. The model has converged with less variance between train and validation loss.

Our approach surpasses the second-best method, DyGLIP, by using the efficient YOLOX detector trained on

multiple datasets for improved generalization. Instead of employing complex structural and spatial attention networks, we utilize a dynamic MPGNN with 0.2 million parameters. Our method achieves superior performance while maintaining lower complexity compared to existing approaches.

V. CONCLUSION

We presented a lightweight graph-based MC-MOT tracker built on tracking by detection paradigm. An anchorless detector (YOLOX) combined with a lightweight feature extraction model (OSNet) helped reduce the model complexity. Also, the generalization capability is enhanced by supervising it on multi-source datasets. Further, the addition of a message-passing graph neural network enhanced the feature representation ability, which helped in reducing missed detections and attained improved tracking performance. The obtained tracking results on popular multi-camera datasets reveal the significance of our approach. The performance of our tracker has improved over the prior arts both for overlapping and non-overlapping FOVs. We anticipate that the tracker's excellent accuracy and ease of use will make it appealing in practical applications. The strongly observed limitation of our study is, since our study strongly depends on appearance features that might affect the tracking performance when the same person appears with change in cloth pattern across multiple cameras is the limitation of our study.

The multi-camera multi-person tracking is conducted by proposing the dynamic graph neural network in a two-stage approach. The following future directions can be explored on our method.

- In the presented work, we consider the appearance features extracted from the message-passing neural network. Since the cross-camera has drastic changes in illumination, person viewpoint, and non-overlapping scenarios, the complete scene itself is different. Therefore, combining some other features, such as IoU, along with the appearance feature, would enhance the tracking performance further.
- The presented method can be applied to single-camera multi-person tracking by training and testing it on MPT challenge benchmarks.
- Graph neural networks have gained attention in many computer vision tasks. Since our method is an edge prediction task which is more like a classification task. This method can be considered to solve any classification problems (e.g., Face recognition).

REFERENCES

[1] J. Ferryman and A. Shahrokni, "PETS2009: Dataset and challenge," in *Proc. 12th IEEE Int. Workshop Perform. Eval. Tracking Surveill.*, Dec. 2009, pp. 1–6.

[2] Z. Ma, X. Wei, X. Hong, and Y. Gong, "Bayesian loss for crowd count estimation with point supervision," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6141–6150.

[3] D. Liciotti, M. Contigiani, E. Frontoni, A. Mancini, P. Zingaretti, and V. Placidi, "Shopper analytics: A customer activity recognition system using a distributed RGB-D camera network," in *Proc. Int. Workshop Video Anal. Audience Meas. Retail Digit. Signage*. Cham, Switzerland: Springer, 2014, pp. 146–157.

[4] Z. Tang, M. Naphade, M.-Y. Liu, X. Yang, S. Birchfield, S. Wang, R. Kumar, D. Anastasiu, and J.-N. Hwang, "CityFlow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8789–8798.

[5] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 17–35.

[6] W. Chen, L. Cao, X. Chen, and K. Huang, "An equalized global graph model-based approach for multicamera object tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 11, pp. 2367–2381, Nov. 2017.

[7] Y. Huang, Z. Xiao, E. Firkat, J. Zhang, D. Wu, and A. Hamdulla, "Spatio-temporal mix deformable feature extractor in visual tracking," *Expert Syst. Appl.*, vol. 237, Mar. 2024, Art. no. 121377.

[8] A. Sadeghian, A. Alahi, and S. Savarese, "Tracking the untrackable: Learning to track multiple cues with long-term dependencies," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 300–311.

[9] K. Otsuka and N. Mukawa, "Multiview occlusion analysis for tracking densely populated objects based on 2-D visual angles," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2004, p. I.

[10] E. Ristani and C. Tomasi, "Features for multi-target multi-camera tracking and re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6036–6046.

[11] O. Javed, K. Shafique, Z. Rasheed, and M. Shah, "Modeling inter-camera space–time and appearance relationships for tracking across non-overlapping views," *Comput. Vis. Image Understand.*, vol. 109, no. 2, pp. 146–162, Feb. 2008.

[12] M. Bredereck, X. Jiang, M. Körner, and J. Denzler, "Data association for multi-object tracking-by-detection in multi-camera networks," in *Proc. 6th Int. Conf. Distrib. Smart Cameras (ICDSC)*, Oct. 2012, pp. 1–6.

[13] L. Wang, F. Dong, Q. Guo, C. Nie, and S. Sun, "Improved rotation kernel transformation directional feature for recognition of wheat stripe rust and powdery mildew," in *Proc. 7th Int. Congr. Image Signal Process.*, 2014, pp. 286–291.

[14] Z. Zhang, J. Wu, X. Zhang, and C. Zhang, "Multi-target, multi-camera tracking by hierarchical clustering: Recent progress on DukeMTMC project," 2017, *arXiv:1712.09531*.

[15] N. Jiang, S. Bai, Y. Xu, C. Xing, Z. Zhou, and W. Wu, "Online inter-camera trajectory association exploiting person re-identification and camera topology," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 1457–1465.

[16] Y. Xu, X. Liu, Y. Liu, and S.-C. Zhu, "Multi-view people tracking via hierarchical trajectory composition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4256–4265.

[17] K. Fathian, K. Khosoussi, Y. Tian, P. Lusk, and J. P. How, "CLEAR: A consistent lifting, embedding, and alignment rectification algorithm for multiview data association," *IEEE Trans. Robot.*, vol. 36, no. 6, pp. 1686–1703, Dec. 2020.

[18] L. Leal-Taixé, G. Pons-Moll, and B. Rosenhahn, "Branch-and-price global optimization for multi-view multi-target tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1987–1994.

[19] J. Dong, W. Jiang, Q. Huang, H. Bao, and X. Zhou, "Fast and robust multi-person 3D pose estimation from multiple views," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7784–7793.

[20] C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan, and M. Grohe, "Weisfeiler and Leman go neural: Higher-order graph neural networks," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 4602–4609.

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

[22] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *Stat*, vol. 1050, p. 20, Oct. 2017.

[23] D. Singh and R. Srivastava, "Graph neural network with RNNs based trajectory prediction of dynamic agents for autonomous vehicle," *Appl. Intell.*, vol. 52, no. 11, pp. 12801–12816, Sep. 2022.

[24] X. Weng, Y. Wang, Y. Man, and K. M. Kitani, "GNN3DMOT: Graph neural network for 3D multi-object tracking with 2D–3D multi-feature learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6498–6507.

[25] G. Brasó and L. Leal-Taixé, "Learning a neural solver for multiple object tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6246–6256.

[26] J. Li, X. Gao, and T. Jiang, "Graph networks for multiple object tracking," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2020, pp. 708–717.

[27] X. Weng, Y. Yuan, and K. Kitani, "PTP: Parallelized tracking and prediction with graph neural networks and diversity sampling," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4640–4647, Jul. 2021.

[28] J. Zhu, H. Zeng, Y. Du, Z. Lei, L. Zheng, and C. Cai, "Joint feature and similarity deep learning for vehicle re-identification," *IEEE Access*, vol. 6, pp. 43724–43731, 2018.

[29] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7122–7131.

[30] V. Malviya and R. Kala, "Trajectory prediction and tracking using a multi-behaviour social particle filter," *Int. J. Speech Technol.*, vol. 52, no. 7, pp. 7158–7200, May 2022.

[31] J. Yang, H. Ge, S. Su, and G. Liu, "Transformer-based two-source motion model for multi-object tracking," *Appl. Intell.*, vol. 52, no. 9, pp. 9967–9979, Jul. 2022.

[32] J. Yang, H. Ge, J. Yang, Y. Tong, and S. Su, "Online multi-object tracking using multi-function integration and tracking simulation training," *Appl. Intell.*, vol. 52, no. 2, pp. 1268–1288, Jan. 2022.

[33] W. Liu, O. Camps, and M. Sznaier, "Multi-camera multi-object tracking," 2017, *arXiv:1709.07065*.

[34] K. Yoon, Y. Song, and M. Jeon, "Multiple hypothesis tracking algorithm for multi-target multi-camera tracking with disjoint views," *IET Image Process.*, vol. 12, no. 7, pp. 1175–1184, Jul. 2018.

[35] Y. He, X. Wei, X. Hong, W. Shi, and Y. Gong, "Multi-target multi-camera tracking by tracklet-to-target assignment," *IEEE Trans. Image Process.*, vol. 29, pp. 5191–5205, 2020.

[36] Y. Hou, L. Zheng, Z. Wang, and S. Wang, "Locality aware appearance metric for multi-target multi-camera tracking," 2019, *arXiv:1911.12037*.

[37] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg, "Multiple hypothesis tracking revisited," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4696–4704.

[38] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua, "Multiple object tracking using K-shortest paths optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1806–1819, Sep. 2011.

[39] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, "Multicamera people tracking with a probabilistic occupancy map," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 267–282, Feb. 2008.

[40] P. Li, G. Li, Z. Yan, Y. Li, M. Lu, P. Xu, Y. Gu, B. Bai, Y. Zhang, and D. Chuxing, "Spatio-temporal consistency and hierarchical matching for multi-target multi-camera vehicle tracking," in *Proc. CVPR Workshops*, 2019, pp. 222–230.

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

[42] H.-M. Hsu, J. Cai, Y. Wang, J.-N. Hwang, and K.-J. Kim, "Multi-target multi-camera tracking of vehicles using metadata-aided re-ID and trajectory-based camera link model," *IEEE Trans. Image Process.*, vol. 30, pp. 5198–5210, 2021.

[43] X. Yang, J. Ye, J. Lu, C. Gong, M. Jiang, X. Lin, W. Zhang, X. Tan, Y. Li, X. Ye, and E. Ding, "Box-grained reranking matching for multi-camera multi-target tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 3095–3105.

[44] Y. Qian, L. Yu, W. Liu, and A. G. Hauptmann, "ELECTRICITY: An efficient multi-camera vehicle tracking system for intelligent city," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 2511–2519.

[45] Z. He, Y. Lei, S. Bai, and W. Wu, "Multi-camera vehicle tracking with powerful visual features and spatial–temporal cue," in *Proc. CVPR Workshops*, vol. 1, 2019, pp. 203–212.

[46] Y. Tariku Tesfaye, E. Zemene, A. Prati, M. Pelillo, and M. Shah, "Multi-target tracking in multiple non-overlapping cameras using constrained dominant sets," 2017, *arXiv:1706.06196*.

[47] H.-M. Hsu, Y. Wang, J. Cai, and J.-N. Hwang, "Multi-target multi-camera tracking of vehicles by graph auto-encoder and self-supervised camera link model," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. Workshops (WACVW)*, Jan. 2022, pp. 489–499.

[48] S. You, H. Yao, and C. Xu, "Multi-target multi-camera tracking with optical-based pose association," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 8, pp. 3105–3117, Aug. 2021.

[49] C. Liu, Y. Zhang, H. Luo, J. Tang, W. Chen, X. Xu, F. Wang, H. Li, and Y.-D. Shen, "City-scale multi-camera vehicle tracking guided by crossroad zones," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 4124–4132.

[50] Y. He, J. Han, W. Yu, X. Hong, X. Wei, and Y. Gong, "City-scale multi-camera vehicle tracking by semantic attribute parsing and cross-camera tracklet matching," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 2456–2465.

[51] J. Ye, X. Yang, S. Kang, Y. He, W. Zhang, L. Huang, M. Jiang, W. Zhang, Y. Shi, M. Xia, and X. Tan, "A robust MTMC tracking system for AI-city challenge 2021," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 4039–4048.

[52] F. Li, Z. Wang, D. Nie, S. Zhang, X. Jiang, X. Zhao, and P. Hu, "Multi-camera vehicle tracking system for AI city challenge 2022," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 3264–3272.

[53] A. Specker, L. Florin, M. Cormier, and J. Beyerer, "Improving multi-target multi-camera tracking by track refinement and completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 3199–3209.

[54] A. Dehghan, S. M. Assari, and M. Shah, "GMMCP tracker: Globally optimal generalized maximum multi clique problem for multiple object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4091–4099.

[55] A. R. Zamir, A. Dehghan, and M. Shah, "GMCP-tracker: Global multi-object tracking using generalized minimum clique graphs," in *Proc. 12th Eur. Conf. Comput. Vis.*, Florence, Italy. Cham, Switzerland: Springer, 2012, pp. 343–356.

[56] E. Ristani and C. Tomasi, "Tracking multiple people online and in real time," in *Proc. 12th Asian Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 444–459.

[57] Y.-G. Lee, Z. Tang, and J.-N. Hwang, "Online-Learning-Based human tracking across non-overlapping cameras," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 10, pp. 2870–2883, Oct. 2018.

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

[59] K. G. Quach, P. Nguyen, H. Le, T.-D. Truong, C. N. Duong, M.-T. Tran, and K. Luu, "DyGLIP: A dynamic graph model with link prediction for accurate multi-camera multiple object tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 13779–13788.

[60] M. Moghaddam, M. Charmi, and H. Hassanpoor, "A robust attribute-aware and real-time multi-target multi-camera tracking system using multi-scale enriched features and hierarchical clustering," *J. Real-Time Image Process.*, vol. 20, no. 3, p. 45, Jun. 2023.

[61] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang, "Learning generalisable omni-scale representations for person re-identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5056–5069, Sep. 2022.

[62] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 3645–3649.

[63] E. Luna, J. C. SanMiguel, J. M. Martínez, and P. Carballeira, "Graph neural networks for cross-camera data association," 2022, *arXiv: 2201.06311*.

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

[65] T. Xiao, S. Li, B. Wang, L. Lin, and X. Wang, "Joint detection and identification feature learning for person search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3376–3385.
[66] L. Zheng, H. Zhang, S. Sun, M. Chandraker, Y. Yang, and Q. Tian, "Person re-identification in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3346–3355.
[67] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 304–311.
[68] S. Bilakeri and K. Kotegar, "Strong baseline with auto-encoder for scale-invariant person re-identification," in *Proc. Int. Conf. Distrib. Comput., VLSI, Electr. Circuits Robot.*, Oct. 2022, pp. 1–6.
[69] S. Bilakeri and K. A. Kotegar, "Combined visual and spatial–temporal information for appearance change person re-identification," *Cogent Eng.*, vol. 10, no. 1, Dec. 2023, Art. no. 2197695.
[70] C. Li, J. Li, Y. Xie, J. Nie, T. Yang, and Z. Lu, "Calibration-free cross-camera target association using interaction spatiotemporal consistency," *IEEE Trans. Multimedia*, vol. 25, pp. 6105–6120, 2023.
[71] Y. Gan, R. Han, L. Yin, W. Feng, and S. Wang, "Self-supervised multi-view multi-human association and tracking," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 282–290.
[72] D. M. H. Nguyen, R. Henschel, B. Rosenhahn, D. Sonntag, and P. Swoboda, "LMGP: Lifted multicut meets geometry projections for multi-camera multi-object tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 8856–8865.
[73] Y. Cai and G. Medioni, "Exploring context information for inter-camera multiple target tracking," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Mar. 2014, pp. 761–768.

SHAVANTREVVA BILAKERI received the bachelor's and master's degrees from Visvesvaraya Technological University, Karnataka, in 2012 and 2014, respectively. She is currently an Assistant Professor with the Department of Data Science and Computer Applications, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India. Her research interests include deep learning, image processing, artificial intelligence, and machine learning.

KARUNAKAR A. KOTEGAR (Senior Member, IEEE) received the B.Sc. and M.C.A. degrees from Karnataka University, Karnataka, India, in 1995 and 1998, respectively, and the Ph.D. degree from Manipal Academy of Higher Education (MAHE), Manipal, Karnataka, in 2009. He is currently the Director of the International Collaborations, MAHE, and a Professor with the Department of Data Science and Computer Applications, Manipal Institute of Technology, MAHE. His research interests include image/video processing and communications, scalable video coding, media aware network elements, multi-view video coding, scalable video over peer-to-peer networks, error resilient and concealment for scalable video, stereo vision, and image and video forensics.