Leveraging Future Trajectory Prediction for Multi-Camera People Tracking

Yuntae Jeon
Department of Global Smart City
Sungkyunkwan University, South Korea
jyt0131@g.skku.edu

Dai Quoc Tran
Global Engineering Insitute for Ultimate Society
Sungkyunkwan University, South Korea
daitran@skku.edu

Minsoo Park
Sungkyun AI Research Institute
Sungkyunkwan University, South Korea
pms5343@skku.edu

Seunghee Park *
School of Civil, Architectural Engineering and Landscape Architecture
Sungkyunkwan University, South Korea
shparkpc@skku.edu

Abstract

Artificial intelligence-based surveillance system, one of the essential systems for smart cities, plays a critical role in ensuring the safety and well-being of individuals. In this paper, we propose a real-time, low-computation cost Multi-Camera Multi-Target (MCMT) tracking system for people, leveraging deep-learning-based trajectory prediction with spatial-temporal information and social information. By predicting people's future trajectories, our algorithm effectively handles object occlusion problems and maintains accurate tracking while keeping computational costs low. Our approach addresses object occlusion without relying on computationally expensive re-identification, and improves MCMT tracking performance using graph-based tracklet representation, and spectral clustering. As a result, our proposed approach is tested on the 2023 AI City Challenge Track 1 test dataset, automatically generated on the NVIDIA Omiverse Platform, our method achieves an IDF1 score of 0.6171 and real-time performance at 27.6 FPS. Code and pre-trained models are publicly available at https://github.com/yuntaeJ/SCIT-MCMT-Tracking.

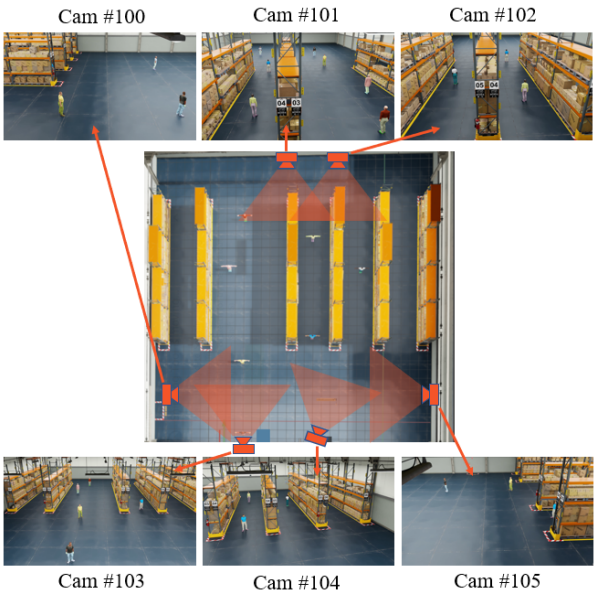

Figure 1. **Dataset visualization of MCMT tracking**. The central global map displays the blueprint of the site, with six distinct camera views (Camera #100, #101, #102, #103, #104, and #105). MCMT tracking task is allocate a consistent ID to one object across the different camera perspectives.

1. Introduction

In recent years, the rapid development of artificial intelligence (AI) has enabled it to address the challenges of urbanization and improve the quality of life for citizens. Especially for surveillance systems, where the data from multiple cameras is vast and manual monitoring of such voluminous data is challenging. Consequently, Multi-Camera Multi-Target (MCMT) tracking becomes an attractive re-

*Corresponding author

search problem and plays a crucial role in ensuring the safety and well-being of individuals in a variety of environments, including urban areas, industrial sites, and public spaces. In the context of smart cities, the MCMT system can be utilized for several purposes, such as monitoring pedestrian traffic, optimizing transportation systems, and detecting potential security threats. In addition, MCMT tracking can be performed through the association of detection and tracking results obtained from Single-Camera Multi-Target (SCMT) tracking with data from multiple sensors in real-time. The solution of data integration and association from multiple sensors in real-time is essential for leveraging AI in solving urban problems.

MCMT-based people tracking has garnered significant attention over recent years, with numerous studies [3, 15, 19, 29, 31, 33] focusing on this subject, particularly through the AI City Challenge [23, 24]. Prior research on MCMT has generally followed the procedures, whereby object detectors are utilized to detect objects, and inter-camera tracking is performed using motion information and appearance information of the detected objects for SCMT tracking. Then, proposed methods in prior research typically involve inter-camera association through motion and appearance information.

However, when objects are overlapped with each other, or when an object is partially occluded by another obstacle, there are still challenges that are hard to solve in some cases such as: 1) Appearance information of the different objects can be too similar within inter-camera views, 2) Appearance information can vary significantly between intra-camera views, and 3) Computational cost of the re-identification (ReID) model. The first case mainly occurs in industrial environments where people wear similar uniforms while working. In such cases, the appearance information among objects is almost identical, rendering the ReID model ineffective. The second case arises from the different angles and detection backgrounds resulting from the installation positions of cameras. Consequently, the appearance information extracted by the ReID model from one view can differ from that extracted by another, making the association even more challenging. To address these issues, researchers have proposed various solutions, including advanced detection and ReID models, transformer-based ReID [14], graph neural networks-based ReID [32], and unsupervised learning-based ReID [10]. However, these approaches often come with trade-offs between accuracy and efficiency, making it challenging to achieve real-time which is mentioned in the third problem.

In this paper, we propose a real-time trajectory prediction-based multi-camera people-tracking method for real-world applications. Our approach uses a lightweight trajectory prediction model consisting of a few convolution neural network (CNN) layers without any ReID models. We were inspired by experiments conducted in ByteTrack [40], comparing two different types of similarity metrics between Intersection-over-Union(IoU) and ReID on the MOT17 dataset. In fact, the IDF1 [30] score was higher when ReID was used, but using only IoU resulted in the highest MOTA [4] score and was about 2.5 times faster than ReID-based tracking.

Our method builds upon the ByteTrack [40] platform, which was initially designed for tracking every detection box with tracklets and utilizes similarities to recover occluded objects and filter out background detections. We also utilize Social-Implict [22] for trajectory prediction in MCMT tracking. We adapt our method to the Multi-Camera People Tracking synthetic dataset (Fig.1) which is generated by the NVIDIA Omniverse Platform and demonstrate its effectiveness in assigning consistent identities to people across different cameras and maintaining accuracy and inference rate.

Our main contributions can be summarized as follows:

1. We propose a method that utilizes temporal motion information to predict future trajectories, which enhances tracking performance for multi-camera multi-object tracking systems.

2. We propose a data association method for integrating single-camera multi-object tracking results across multiple cameras, leading to improved tracking consistency.

3. We demonstrate that our multi-camera-based object tracking system runs in real-time with low computational power, making it suitable for edge devices and practical real-world applications, such as smart city monitoring and analysis.

The rest of the paper is organized as follows: an overview of related work is described in Section 2. Section 3 introduces our proposed framework in detail. In Section 4, we demonstrate sufficient experiments of our method on track 1 of CVPR 2023 7th AI City Challenge [25]. Finally, we present the conclusion in Section 5.

2. Related Work

2.1. Object Detection

Object detection, a crucial component of object tracking, has evolved with the development of one-stage and two-stage detectors [42]. One-stage detectors, such as YOLO series [11, 27, 28], RetinaNet [20], and CenterNet [8], offer a balance between accuracy and speed, making them suitable for real-time tracking applications. Two-stage detectors, like Faster R-CNN [12], provide higher accuracy but at the expense of increased processing time. In this

work, we use the YOLOX [11] model, an anchor-free detector, which simplifies the model and reduces the number of design parameters that require heuristic tuning, making it more straightforward during training and decoding phases.

2.2. Person Re-Identification

Person ReID [10,13] also can be used for object tracking, playing a role in solution for object overlap to compare the appearance information of objects to ensure that the object ID does not change before and after overlapping occurs. Recently, the transformer-based model [14] has improved person ReID considerably. However, in our paper, we do not use ReID model due to its computational demands, instead, we use a deep-learning-based trajectory prediction model for leveraging motion information. This approach not only replaces the traditional role of ReID by solving the occlusion problem but also increases the efficiency of the model, rate and computational cost.

2.3. Trajectory Prediction

Some researchers [9, 38] utilize deep-learning-based trajectory prediction for object tracking. Recently, trajectory prediction has emerged as a critical research area, with applications spanning autonomous vehicles, robotics, and crowd analysis, among others. Several approaches have been proposed to address this challenge, including deep learning-based techniques such as Long Short-Term Memory (LSTM) networks [2, 34], Convolutional Neural Networks (CNNs) [22], and Graph Neural Networks (GNNs) [18, 21]. In trajectory prediction research, it is common to incorporate three types of information: scene information obtained from segmenting the background map, spatial-temporal information derived from individual movement paths, and social information that considers interactions between people. This comprehensive approach helps to create more accurate and realistic trajectory predictions by accounting for environmental factors, personal preferences, and interpersonal dynamics. In our research, we utilize a Social-Implicit [22] based on CNNs, which only take into account spatial-temporal information and social information. This approach allows us to focus on the key factors influencing human movement and interactions, leading to effective trajectory predictions.

2.4. Single-Camera Object Tracking

Single-Camera object tracking methods can be divided into tracking-by-detection and joint-detection-tracking methods. The former, such as SORT [5] and DeepSORT [37], detect objects first and then associate them based on appearance and motion cues. These methods have been dominant in single-camera multi-target tracking tasks, but are limited in accuracy. The latter, such as FairMOT [41] and ByteTrack [40], incorporate appearance embedding or motion prediction into detection frameworks, offering comparable performance with low computational costs. However, there can be competition between different components that lower the upper bound of tracking performance. Recently, BotSORT [1] with ReID surpassed ByteTrack by utilizing both motion and appearance information. We follow the ByteTrack platform replacing the Kalman Filter [16] algorithm to a deep-learning-based trajectory prediction model and add the Hungarian Algorithm [17] for ID matching based on IoU distance.

2.5. Multi-Camera Object Tracking

Multi-Camera object tracking research has progressed significantly by employing techniques such as object detection, appearance feature extraction for ReID, and inter-camera tracklets matching. The main issue about MCMT is a tracklets clustering problem and focuses on reducing the search space. Hierarchical trajectory composition approach [39] that utilizes multiple mutually complementary 2D and 3D cues, such as ground occupancy consistency, appearance similarity, and motion coherence. Others suggest Tracklet-Plane matching [26] approach enhances multiple object tracking by organizing temporally-related object detections into planes and reducing confusion among similar tracklets. Graph-based matching [7] approach creates a graph model between multiple tracklets from different cameras, optimizing for MTMC tracking solution. Our research uses a graph model to represent the tracklets from multiple cameras, then Spectral Clustering [35] is executed on the graph for inter-camera tracklets matching.

3. Methodology

We propose a real-time and computation-effective MCMT tracking of people. Firstly, we detect bounding boxes (bbox) and keypoints using an object detector from every single camera. Then, we can obtain spatial information about the position of each object and temporal information from continuous frames captured by the cameras. Our method also considers social information that can affect each object by taking into account the spatial-temporal information between the objects. Finally, we integrate tracklets from different single-camera in real-time as an online method, which utilizes only motion information for MCMT tracking.

3.1. Object Detection

In order to perform object tracking, object detection results are first required. We utilized YOLOX [11] model, which is an anchor-free based object detection model based on the 1-stage method, to enable the model to run in real-time with low computation cost. The advantage of the YOLOX model over other object detection models is its ability to achieve high detection accuracy when objects are

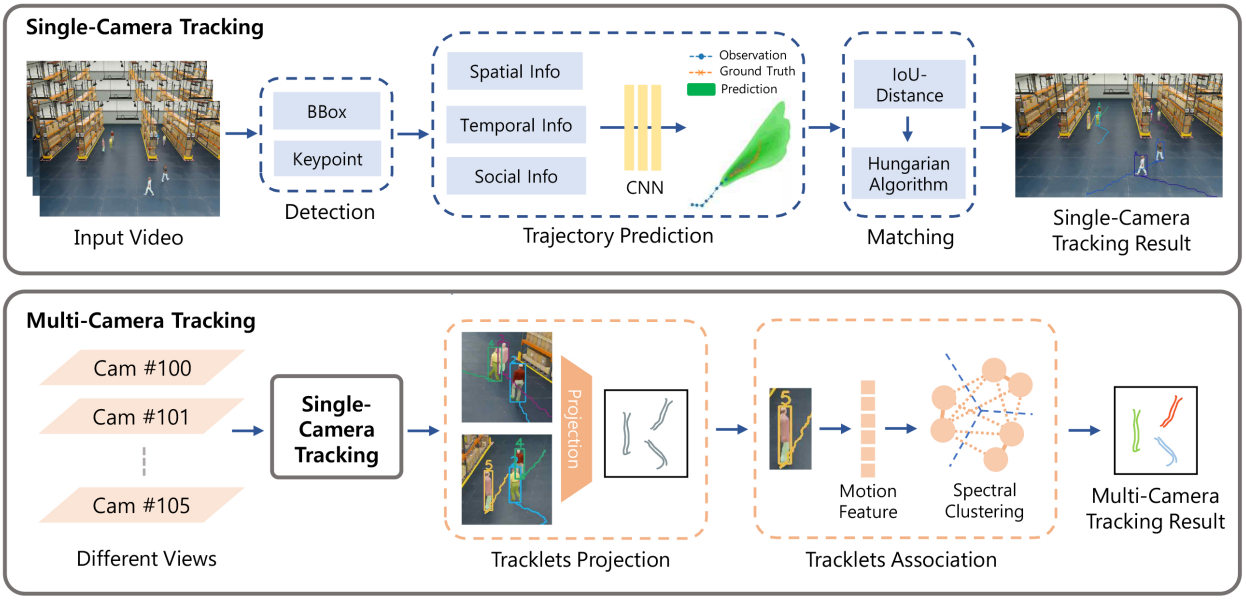

Figure 2. **Proposed Pipeline.** Our MCMT tracking is performed in the following sequence. First, for each camera with different views, objects are detected through detection, and trajectory prediction is performed using the spatio-temporal info and social info of the detection results. Then, Single-Camera Tracking is performed based on the IoU Distance. Next, the tracklets are projected onto the global map, and finally, the tracklets are associated with each other using motion features-based spectral clustering, completing the MCMT process.

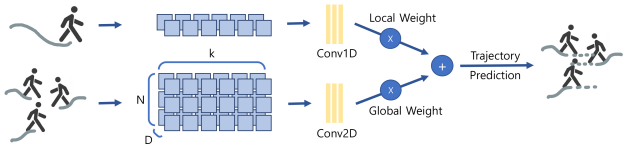

Figure 3. **Trajectory Prediction Process.** Where D is the dimension of input(x-y coordinates), N is the number of objects, and k is the sequence of motion information. The first step involves extracting spatio-temporal information for individual objects, while the second step focuses on extracting spatio-temporal information for multiple objects, which represents social information. By considering both local and global weights, we can effectively predict future trajectories. Redraw from Social-Implicit [22].

overlapped each other. However, the most important thing in object detection tasks is the quality and diversity of training data, since a well-curated and representative dataset is crucial for achieving optimal performance, regardless of the chosen model architecture. As a result, during this challenge, we refined the training dataset provided by the NVIDIA Omniverse platform. This was necessary because the automatically generated dataset contained noisy data, including occluded objects or partially visible body parts like heads or legs only.

Furthermore, in addition to detecting bounding boxes, we also use a person keypoints detection model for estimating the position of a person's ankles. If the human keypoint detection results show only up to the waist or shoulder area instead of the feet, we can estimate the position of the legs by utilizing the proportional relationship between the detected keypoints. In order to run the model in real-time, we used the YOLOv7 [36] pose estimation model, and since the keypoint training dataset was not provided in the competition dataset, a pre-trained model is utilized.

3.2. Trajectory Prediction

In order to assign IDs to detected objects in consecutive frames, it is essential to have feature values for the respective objects. In this paper, we propose using a deep-learning-based trajectory prediction model to perform this task based on the IoU distance values between motion features. We propose a novel approach to address the object occlusion problem in object tracking tasks by using deep learning-based trajectory prediction instead of ReID models, possible for real-time application. We utilize the Social-Implicit [22] model for this purpose, which performs trajectory prediction using both spatial-temporal information and social information.

Spatio-Temporal Information, based on motion information of an object in space during continuous frames, significantly impacts regression tasks with sequences as inputs and outputs. While observing from frame t_1 to t_{obs}, the motion states of one object, denoted as $m_{t_1:t_{\text{obs}}} = \{m_t | t \in [t_1, \cdots, t_{\text{obs}}]\}$, where $m_t \in \mathbb{R}^{D \times obs}$. The symbol D represents the dimension of the input motion state. In

this context, it pertains to the x-y coordinates of an object's position, making $D = 2$. From the observed motion information, firstly we extract the key motion information from continuous frames. Therefore, we cluster m_t based on the pre-defined cluster size τ, the number of near frames. When clustering, we apply Fast Fourier Transformation (FFT) for extracting key motion information to remove noise. Perform the FFT on the clustered $m = (m_1, m_2, ..., m_\tau)$:

$$F_x(x) = \text{FFT}(x), \quad F_y(y) = \text{FFT}(y) \tag{1}$$

where x, y are x-y coordinates at m_t. Then define the binary filter, which removes high-frequency components and keeps only the low-frequency components:

$$\text{Filter} = \{f_i \mid f_i = \begin{cases} 1, & \text{if } i \leq \tau \cdot r \\ 0, & \text{otherwise} \end{cases}\} \tag{2}$$

where f_i is the binary value of the filter array at index i, τ is the total number of frequency components same as the pre-defined cluster size, and r is also a pre-defined parameter about cutoff frequency ratio between 0 and 0.5. Then apply the filter by element-wise multiplication. This operation removes the high-frequency components from the transformed data. Perform the Inverse Fast Fourier Transform (IFFT) on the filtered frequency data to obtain the filtered time-domain data (x-y coordinates). Since the result of the IFFT can be complex, we take the real part of the result:

$$\begin{aligned} filtered\_x &= \text{real}(\text{IFFT}(F_x(x) \cdot \text{Filter}) \\ filtered\_y &= \text{real}(\text{IFFT}(F_y(y) \cdot \text{Filter}) \end{aligned} \tag{3}$$

where $filtered\_x$ and $filtered\_y$ are the key motion information of one cluster. After clustering, each object's motion information can express to $m_t \in \mathbb{R}^{D \times k}$, k is from the length of obs divided by τ. Fig.3 illustrated clustered motion information of the single object. To extract the spatio-temporal information of a single object we use 2 CNN layers as follows:

$$\begin{aligned} v_{\text{feat}} &= \text{ReLU}(\text{Conv1D}(m_t) \\ v_{\text{res}} &= \text{Conv1D}(m_t) \\ v_{\text{spatial}} &= v_{\text{feat}} + v_{\text{res}} \\ v_{\text{res}} &= \text{Conv1D}(v_{\text{spatial}}) \\ v_{\text{spatio-temporal}} &= \text{Conv1D}(v_{\text{spatial}}) + v_{\text{res}} \end{aligned} \tag{4}$$

where $v_{\text{spatio-temporal}}$ is spatio-temporal information of one object, which would be utilized for trajectory prediction fused with social information explained in the next section.

Social information refers to the data related to the interactions and relationships between multiple objects within a given environment. The motion information of multi objects can be expressed to $M_t \in \mathbb{R}^{D \times N \times k}$, where N means the number of objects. To extract the relationships between multiple objects we execute 2D convolution to M_t:

$$\begin{aligned} V_{\text{feat}} &= \text{ReLU}(\text{Conv2D}(M_t) \\ V_{\text{res}} &= \text{Conv2D}(M_t) \\ V_{\text{spatial}} &= V_{\text{feat}} + V_{\text{res}} \\ V_{\text{res}} &= \text{Conv2D}(V_{\text{spatial}}) \\ V_{\text{spatio-temporal}} &= \text{Conv2D}(V_{\text{spatial}}) + V_{\text{res}} \end{aligned} \tag{5}$$

where $V_{\text{spatio-temporal}}$ is social information about total N objects, and it could be fused with each object's spatio-temporal information to predict the future trajectory:

$$V = w_g \cdot v_g + w_l \cdot v_l \tag{6}$$

where w_g and w_l are weights of global and local, respectively, v_g and v_l are spatio-temporal information of global and local, and V represents the predicted future positions of objects, $M_{t_k+1:t_k+\text{pred}}$.

Trajectory Prediction based Tracking utilizes motion information of target objects. Recently, ByteTrack [40] proposed an object detection threshold-based two-step matching algorithm for tracking every detection box with tracklets and utilizes similarities to recover occluded objects. We apply the ByteTrack platform replacing the Kalman Filter [16] algorithm-based results to V from our proposed method. Based on the predicted future positions, V, we can calculate IoU distance. The IoU distance is an effective measure for evaluating the similarity between two bounding boxes, as it quantifies the ratio of the intersection area to the union area of the boxes. In our method, we predict the future positions of the objects and compute the IoU distance between the predicted and actual bounding boxes in the subsequent frames. The IoU distance is then utilized as a cost function for associating objects across frames, which is given by the following equation:

$$IoU(A, B) = \frac{area(A \cap B)}{area(A \cup B)} \tag{7}$$

where A and B are the bounding boxes of two objects being compared. To ensure proper assignment of object identities during occlusions, we employ the Hungarian algorithm [17] for data association, which is a combinatorial optimization algorithm that solves the assignment problem in polynomial time. Given a cost matrix representing the IoU distances between the predicted and actual bounding boxes, the Hungarian algorithm finds the optimal assignment that minimizes the overall cost. The cost matrix can be formulated as follows:

$$C_{ij} = 1 - IoU(P_i, D_j) \tag{8}$$

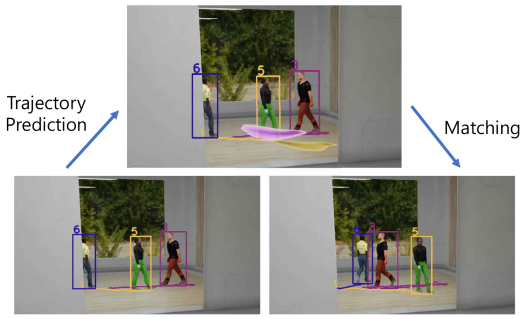

Figure 4. **Visualization of trajectory prediction based matching when object occlusion occurs**. First, our model predicts the future trajectories for each object. Then, after overlap occurs, we check whether the ID is properly maintained without changing.

where C_{ij} is the cost associated with assigning the predicted bounding box P_i to the detected bounding box D_j. The Hungarian algorithm operates on this cost matrix to find the optimal assignment that minimizes the overall cost, which can be represented as:

$$\min_{A} \sum_{i=1}^{n} \sum_{j=1}^{m} C_{ij} A_{ij} \tag{9}$$

subject to the constraints:

$$\sum_{i=1}^{n} A_{ij} = 1, \quad \sum_{j=1}^{m} A_{ij} = 1 \tag{10}$$

where A_{ij} is a binary variable indicating whether the predicted bounding box P_i is assigned to the detected bounding box D_j, and n and m are the number of predicted and detected bounding boxes, respectively. By employing the Hungarian algorithm, our method ensures correct identity assignment for objects even after occlusions, which significantly improves the tracking consistency in multi-camera systems.

3.3. Multi-camera Object Tracking

We suggest MCMT tracking system based on SCMT tracking results. Our approach involves extracting tracklets about all objects from each camera view, projecting them onto a global map, and creating a graph representation based on each tracklet's motion features. Subsequently, we apply spectral clustering for tracklet association, as expressed in Fig. 2.

3.3.1 Tracklet Projection

We aim to project SCMT tracklets onto a global map using a homography matrix. The homography matrix is utilized to establish a relationship between the coordinates in the camera frame and the global map, allowing us to extract global tracklet positions from the SCMT tracklets. Given a point $\mathbf{p}_1'$ in the camera frame and its corresponding point $\mathbf{p}_2'$ in the global map, we compute the homography matrix $\mathbf{H}$ and apply the transformation as follows:

$$\mathbf{p}_2' = \mathbf{H} \begin{bmatrix} c_x & b_y & 1 \end{bmatrix} \tag{11}$$

where c_x and b_y represent the center and bottom y-coordinate of the bounding box, respectively. Finally, we normalize the resulting point $\mathbf{p}_2'$ to extract the global tracklet position in the global map:

$$\mathbf{p}_2 = \frac{\mathbf{p}2'}{p2'z} \tag{12}$$

By employing this method, we can effectively project SCMT tracklets onto a global map, enabling a comprehensive analysis of objects' movements in a unified coordinate system.

3.3.2 Tracklet Association

We elaborate on the graph-based tracklet representation and the association process using spectral clustering. We first construct an affinity graph $G = (V, E)$, where V denotes the set of tracklets and E represents the weighted edges based on motion features between tracklets. The affinity matrix A is defined as:

$$A_{ij} = \exp(-\frac{|F_i - F_j|^2}{\sigma^2}) \tag{13}$$

where F_i and F_j represent the motion features of tracklets i and j, and σ is a scaling factor. Spectral clustering is applied to partition the affinity graph into disjoint clusters. To do this, we compute the graph Laplacian L:

$$L = D - A \tag{14}$$

where D is the degree matrix, with $D_{ii} = \sum_j A_{ij}$. Then, we find the k smallest eigenvectors of L to form a matrix $U \in \mathbb{R}^{n \times k}$, where n is the number of tracklets. Next, we normalize the rows of U to form a matrix T:

$$T_{ij} = \frac{U_{ij}}{|U_i|} \tag{15}$$

Finally, we apply the k-means algorithm to cluster the rows of T into k clusters, where each cluster corresponds to a unique object tracked across multiple cameras. This approach enables a robust and efficient association of tracklets across different camera views, resulting in improved tracking performance in multi-camera systems.

Figure 5. **Visualization of SCMT tracking according to object occlusion.** Frame t_1 shows the situation before objects overlap, t_2 shows the moment when the overlap occurs, and frame t_3 shows the situation after the overlap has taken place. (a) Our method demonstrates solving the object overlapping issue that (b) Bytetrack [40] could not resolve previously, while maintaining real-time processing speed without ReID.

4. Experiments

4.1. Dataset

2023 AI City Challenge [25] dataset for track 1, which includes data from 1,440 minutes of video captured by 129 cameras across 22 indoor sites, with 10 sites for training, 5 for validation, and 7 for testing. The dataset contains both real-world data from various settings such as warehouses and buildings, as well as synthetically generated data from multiple indoor environments created using the NVIDIA Omniverse Platform. All video feeds are high-resolution 1080p at 30 frames per second. For object detection training, we preprocessed the training dataset, extracted images, and filtered out noisy images based on certain rules, resulting in 15,539 images with 61,314 bounding boxes. One of the difficulties with the provided dataset is that almost every frame and object is labeled, and in some cases, the object size is so minute that it appears impossible for humans to recognize. As a result, we perform extensive preprocessing on all frames and objects, filtering out minute and difficult-to-recognize objects using a threshold. For trajectory prediction training, we used the full train dataset and employed an FFT-based method for key motion information extraction.

4.2. Evaluation Metric

The IDF1 score evaluates the multi-camera object tracking system by measuring trajectory consistency within the camera network, calculated as:

$$IDF1 = \frac{2 \cdot IDTP}{2 \cdot IDTP + IDFP + IDFN} \tag{16}$$

where $IDTP$ is the count of true positive identities, $IDFP$ is the quantity of false-positive identities, and $IDFN$ is the total of false negative identities.

4.3. Implementation Detail

The training process was conducted on two NVIDIA A6000 GPUs, and the testing was performed on one A6000 GPU. PyTorch 1.11.0 was used as the deep learning framework. For object detection, we used YOLOX [11] to generate bounding boxes. We used the COCO-pre-trained model with a threshold of 0.1 and trained it for 300 epochs with a learning rate of 0.001 and a batch size of 8 on resized im-

ages (1333×800 pixels). We trained the model using the MMDetection [6] toolbox. For keypoint estimation, we directly used the YOLOv7 [36] pose-estimation model with the pre-trained model. Finally, for the trajectory prediction, we use Social-Implicit [22] model, and trained it for 50 epochs with a learning rate of 0.01 and a batch size of 128. We utilize ByteTrack [40] for SCMT tracking and based on that result, finally, we run our MCMT tracking system for this challenge.

4.4. Quantitive Result

We test our proposed MCMT tracking system on the 2023 AI City Challenge [25] test dataset, and the results can get an IDF1 score of 0.6171, shown in Table 1. We also compare the inference rate based on the test set, shown in Table 2. Our method demonstrates the advantage of incorporating trajectory prediction for object tracking, achieving real-time performance at 27.6 FPS. This is faster than other methods that use ReID, such as DeepSort and BotSort, which operate at 15.6 FPS and 13.6 FPS, respectively. A key benefit of our proposed method is its ability to address the object occlusion problem without relying on computationally expensive ReID techniques. By predicting object trajectories, our algorithm can effectively handle occlusions and maintain accurate tracking, while keeping the computational cost relatively low. Table 3 proves that our trajectory prediction (TP) technique can address the issues present in the baseline ByteTrack model. Finally, when we add an FFT component to our method, resulting in an IDF1 score of 0.6171.

Table 1. The results of 2023 AI City Challenge Track1.

Rank	Team	IDF1
1	Team 6	0.9536
2	Team 9	0.9417
3	Team 41	0.9331
...	...	...
13	**Team 20 (Ours)**	**0.6171**
14	Team 64	0.4660
15	Team 191	0.4546

4.5. Qualitive Result

Figure 5 demonstrates that our proposed method is capable of resolving the ID switching issue that occurs in overlapping situations when using the ByteTrack [40]. Up until now, many studies have addressed this problem by employing ReID-based solutions such as BotSORT [1]. However, our research shows that by utilizing trajectory prediction, it is possible to solve this issue without the need for a ReID model, thereby maintaining real-time inference speeds and enhancing object tracking performance.

Table 2. The comparison of SCMT tracking models.

Model	module			FPS
	Motion	ReID	TP	
SORT [5]	✓			33.2
DeepSORT [3]	✓	✓		15.6
ByteTrack [40]	✓			33.1
BotSORT [1]	✓	✓		13.6
Ours	✓		✓	27.6

Table 3. The performance of each proposed module.

Model	IDF1	IDP	IDR
baseline	0.4752	0.4989	0.4537
+ TP	0.5940	0.6156	0.5738
+ TP + FFT **(Ours)**	0.6171	0.6392	0.5965

5. Conclusion

In this paper, we have presented a novel Multi-Camera Multi-Target (MCMT) tracking system based on people's future trajectories prediction. Our approach effectively handles occlusions and maintains accurate tracking while minimizing computational costs by incorporating deep-learning-based trajectory prediction with spatial-temporal information and social information, as well as graph-based tracklet representation and spectral clustering.

Our proposed system has been evaluated on the 2023 AI City Challenge [25] Track1 test dataset and demonstrated its superiority over baseline models and ReID-based methods such as DeepSort and BotSort, achieving an IDF1 score of 0.6171 and real-time performance at 27.6 FPS. The use of trajectory prediction techniques has proven effective in resolving ID-switching issues in overlapping situations, which have traditionally been addressed by employing ReID-based solutions. Furthermore, our real-time MCMT tracking method, which utilizes trajectory prediction, can play a crucial role in AI-based smart cities by proactively predicting and making decisions for various situations that may arise in real-world scenarios.

Acknowledgements:

This research was supported by a grant [2022-MOIS38-002] from the Ministry of Interior and Safety (MOIS)'s project, and a grant from the Korea government (MSIT) to the National Research Foundation of Korea (NRF) [NRF-2021R1A4A3033128]. In addition, Korea Ministry of Land, Infrastructure and Transport(MOLIT) as Innovative Talent Education Program for Smart City, and BK21 FOUR Project.

References

[1] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking. *arXiv preprint arXiv:2206.14651*, 2022. 3, 8

[2] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016. 3

[3] Pierre Baqué, François Fleuret, and Pascal Fua. Deep occlusion reasoning for multi-camera multi-target detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 271–279, 2017. 2, 8

[4] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008. 2

[5] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. IEEE, 2016. 3, 8

[6] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 8

[7] Weihua Chen, Lijun Cao, Xiaotang Chen, and Kaiqi Huang. An equalized global graph model-based approach for multi-camera object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(11):2367–2381, 2016. 3

[8] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6569–6578, 2019. 2

[9] Tharindu Fernando, Simon Denman, Sridha Sridharan, and Clinton Fookes. Tracking by prediction: A deep generative model for mutli-person localisation and tracking. In *2018 IEEE Winter conference on applications of computer vision (WACV)*, pages 1122–1132. IEEE, 2018. 3

[10] Dengpan Fu, Dongdong Chen, Jianmin Bao, Hao Yang, Lu Yuan, Lei Zhang, Houqiang Li, and Dong Chen. Unsupervised pre-training for person re-identification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14750–14759, 2021. 2, 3

[11] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 2, 3, 7

[12] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 2

[13] Lingxiao He, Xingyu Liao, Wu Liu, Xinchen Liu, Peng Cheng, and Tao Mei. Fastreid: A pytorch toolbox for general instance re-identification. *arXiv preprint arXiv:2006.02631*, 2020. 3

[14] Shuting He, Hao Luo, Pichao Wang, Fan Wang, Hao Li, and Wei Jiang. Transreid: Transformer-based object re-identification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15013–15022, 2021. 2, 3

[15] Yuhang He, Xing Wei, Xiaopeng Hong, Weiwei Shi, and Yihong Gong. Multi-target multi-camera tracking by tracklet-to-target assignment. *IEEE Transactions on Image Processing*, 29:5191–5205, 2020. 2

[16] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960. 3, 5

[17] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 3, 5

[18] Maosen Li, Siheng Chen, Yangheng Zhao, Ya Zhang, Yanfeng Wang, and Qi Tian. Dynamic multiscale graph neural networks for 3d skeleton based human motion prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 214–223, 2020. 3

[19] Peng Li, Jiabin Zhang, Zheng Zhu, Yanwei Li, Lu Jiang, and Guan Huang. State-aware re-identification feature for multi-target multi-camera tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2

[20] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 2

[21] Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14424–14432, 2020. 3

[22] Abduallah Mohamed, Deyao Zhu, Warren Vu, Mohamed Elhoseiny, and Christian Claudel. Social-implicit: Rethinking trajectory prediction evaluation and the effectiveness of implicit maximum likelihood estimation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*, pages 463–479. Springer, 2022. 2, 3, 4, 8

[23] M. Naphade, S. Wang, D. C. Anastasiu, Z. Tang, M. Chang, Y. Yao, L. Zheng, M. Shaiqur Rahman, A. Venkatachalapathy, A. Sharma, Q. Feng, V. Ablavsky, S. Sclaroff, P. Chakraborty, A. Li, S. Li, and R. Chellappa. The 6th ai city challenge. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3346–3355. IEEE Computer Society, June 2022. 2

[24] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Xiaodong Yang, Liang Zheng, Anuj Sharma, Rama Chellappa, and Pranamesh Chakraborty. The 4th ai city challenge. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, page 2665–2674, June 2020. 2

[25] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Yue Yao, Liang Zheng, Mo-

hammed Shaiqur Rahman, Meenakshi S. Arya, Anuj Sharma, Qi Feng, Vitaly Ablavsky, Stan Sclaroff, Pranamesh Chakraborty, Sanjita Prajapati, Alice Li, Shangru Li, Krishna Kunadharaju, Shenxin Jiang, and Rama Chellappa. The 7th AI City Challenge. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2023. 2, 7, 8

[26] Jinlong Peng, Tao Wang, Weiyao Lin, Jian Wang, John See, Shilei Wen, and Erui Ding. Tpm: Multiple object tracking with tracklet-plane matching. *Pattern Recognition*, 107:107480, 2020. 3

[27] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 2

[28] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 2

[29] Pengfei Ren, Kang Lu, Yu Yang, Yun Yang, Guangze Sun, Wei Wang, Gang Wang, Junliang Cao, Zhifeng Zhao, and Wei Liu. Multi-camera vehicle tracking system based on spatial-temporal filtering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4213–4219, 2021. 2

[30] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part II*, pages 17–35. Springer, 2016. 2

[31] Ergys Ristani and Carlo Tomasi. Features for multi-target multi-camera tracking and re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6036–6046, 2018. 2

[32] Yantao Shen, Hongsheng Li, Shuai Yi, Dapeng Chen, and Xiaogang Wang. Person re-identification with deep similarity-guided graph neural network. In *Proceedings of the European conference on computer vision (ECCV)*, pages 486–504, 2018. 2

[33] Andreas Specker, Daniel Stadler, Lucas Florin, and Jurgen Beyerer. An occlusion-aware multi-target multi-camera tracking system. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4173–4182, 2021. 2

[34] Chaofan Tao, Qinhong Jiang, Lixin Duan, and Ping Luo. Dynamic and static context-aware lstm for multi-agent motion prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI*, pages 547–563. Springer, 2020. 3

[35] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17:395–416, 2007. 3

[36] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv preprint arXiv:2207.02696*, 2022. 4, 8

[37] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE, 2017. 3

[38] Yang Xing, Chen Lv, and Dongpu Cao. Personalized vehicle trajectory prediction based on joint time-series modeling for connected vehicles. *IEEE Transactions on Vehicular Technology*, 69(2):1341–1352, 2019. 3

[39] Yuanlu Xu, Xiaobai Liu, Yang Liu, and Song-Chun Zhu. Multi-view people tracking via hierarchical trajectory composition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4256–4265, 2016. 3

[40] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*, pages 1–21. Springer, 2022. 2, 3, 5, 7, 8

[41] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International Journal of Computer Vision*, 129:3069–3087, 2021. 3

[42] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu. Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, 30(11):3212–3232, 2019. 2