MSMCT: Multi-State Multi-Camera Tracker

Behzad Bozorgtabar, and Roland Goecke, *Member, IEEE*

***Abstract*—Visual tracking of multiple persons simultaneously is an important tool for group behaviour analysis. In this paper, we demonstrate that multi-target tracking in a network of non-overlapping cameras can be formulated in a framework, where the association among all given target hypotheses both within and between cameras is performed simultaneously. Our approach helps to overcome the fragility of multi-camera based tracking, where the performance relies on the single-camera tracking results obtained at input level. In particular, we formulate an estimation of the target states as a multi-state graph optimisation problem, in which the likelihood of each target hypothesis belonging to different identities is modelled. In addition, we learn the target-specific model to improve the similarity measure among targets based on the appearance cues. We also handle the occluded targets when there is no reliable evidence for the target's presence and each target trajectory is expected to be fragmented into multiple tracks. An iterative procedure is proposed to solve the optimisation problem, resulting in final trajectories that reveal the true states of the targets. The performance of the proposed approach has been extensively evaluated on challenging multi-camera non-overlapping tracking datasets, in which many difficulties such as occlusion, viewpoint and illumination variation are present. The results of systematic experiments conducted on a large set of sequences show that the proposed approach outperforms several state-of-the-art trackers.**

***Index Terms*—Multi-camera target tracking, multiple states graphical model, target-specific metric learning.**

I. Introduction

MULTI-TARGET tracking across a camera network involving multiple non-overlapping cameras has recently drawn great interest due to its vital importance in many computer vision applications such as visual surveillance and group behaviour analysis. In this scenario, the aim is to track and recover spatio-temporal trajectories for a number of targets when they move within or across cameras.

Most of the recent approaches that aim to solve this problem follow two main directions: (1) tracking multiple targets, such as people, within each camera view, and (2) inter-camera tracking, such as maintaining the identity of the people when they disappear from the field-of-view (FOV) of one camera and reappear in the FOV of another camera. Intra-camera tracking requires correspondence among the target hypotheses in an individual camera while considering a spatio-temporal smoothness constraint between neighbouring target candidates. On the other hand, inter-camera tracking matches the individual target hypotheses over disjoint cameras, which is more challenging than intra-camera tracking as there are more variations in target viewpoints, illumination conditions and background. Moreover, in the absence of camera calibrations and information regarding the projected target positions in a non-overlapping camera setting, the appearance cue is the only information that can be used for the target correspondence.

To address these problems, existing methods mainly focus on developing the data association techniques over trajectories obtained from individual cameras. However, a main downside to such schemes is that their performance largely depends on the performance of a single-camera target tracker. If the single-camera tracker fires many of identity switches and *false alarms*, the cross-view multi-target tracking fails consequently. In particular, for the challenging dataset [1] (see Fig. 1), the state-of-the-art trackers [2], [3] are prone to ID-switches and other difficulties, such as occlusions and illumination changes. Therefore, the target trajectory is expected to be fragmented within a single-camera and it causes failure in the multi-camera data association among the target tracks (see Fig. 2). This ambiguity in association can also be caused by similar appearance of targets across all cameras, which obviously decreases the accuracy of the corresponding tracker.

In this paper, we present a new multi-camera tracker, which integrates intra-camera data association and inter-camera association among targets in a unified framework. The proposed approach exploits the joint information between and within individual cameras, resulting in similar cross-view target tracks to be extracted from inherently noisy and fragmented target tracks.

More specifically, we formulate the problem of tracking multiple targets in the cross-view cameras as a *multi-state* graph optimisation problem, in which each graph node encodes different target states (identities) and the graph edges represent the similarities between different target hypotheses. The edge costs include two different affinity metrics: (1) a within a single-camera affinity model, which utilises the optical flow trajectories as a more reliable similarity measure between target hypotheses across time, and (2) an inter-camera target appearance model, where a target-specific metric learning model is proposed to obtain effective appearance cues for reliable association between different tracks. Here, the higher order correspondences between the observations (targets' hypotheses across cameras) are considered.

The input to our tracker (graph nodes) in every camera view is an initial low-level target track instead of sparse target detection responses. This helps the tracker to not only reduce the computational complexity, but also allows integrating the tracks' smoothness in the spatio-temporal domain into the edge cost of the graph. This advantage rectifies noisy detections mostly caused by occlusion or inaccurate pre-trained detectors. In order to obtain the optimisation solution efficiently, an ex-

B. Bozorgtabar is with Vision & Sensing, Human-Centred Technology Research Centre, University of Canberra, Australia.
E-mail: behzad.bozorgtabar@canberra.edu.au

Roland Goecke is with Vision & Sensing, Human-Centred Technology Research Centre, University of Canberra, Australia.
E-mail: roland.goecke@ieee.org

©20xx IEEE. Personal use is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending an email to pubs-permissions@ieee.org.

isting energy minimisation technique is used for the proposed graph inference, which is a variant of *Conditional Random Field* (*CRF*) approaches.

II. Related Work

Most previous approaches to multi-camera tracking comprise of two steps: (i) obtaining a set of target hypotheses in each video frame of individual cameras, such as detections or initial target tracks, and (ii) assignment between target hypotheses across different cameras (*multi-camera data association*). Although there is some related work [4], [5], which addresses both problems (single-camera tracking and data association between cameras) by proposing these two parts in a single framework, the majority of state-of-the-art methods considered these two problems separately and the priority is usually given to the multi-camera data association design. The reason is that in a multi-camera setting, a more sophisticated similarity measure and feature representations are needed to handle relationships between targets. Hence, the related work in both single view tracking and multi-view tracking frameworks are reviewed here.

A. Single View Multi-person Tracking

In comparison with the multi-camera tracking framework, there is more consistency in spatio-temporal features of targets within a single-camera view. However, ambiguities arise in the presence of detection errors (false negatives or extra detections) or when several visually similar targets move closely together. Most recent approaches to single view tracking pursue a *tracking-by-detection* strategy, where the correspondence between the target observations in a set of subsequent frames is considered.

Brendel *et al.* [6] presented a framework to formulate the data association problem as finding the *maximum-weight independent set* (MWIS) of the graph of detections' responses. Shafique *et al.* [7] proposed a *k-partite* complete graph where the associations among targets (graph nodes) are not limited to only two consecutive frames. Some other approaches formulate single view multi-person tracking as a *network flow* problem where a set of object tracks are resolved simultaneously by solving min-cost flow techniques. Zhang *et al.* [8] proposed a global optimal solution for the network flow optimisation using the *push-relabel* algorithm. Berclaz *et al.* [9] presented a globally tracking framework based on the *k-shortest paths* (KSP) algorithm to solve the flow problem.

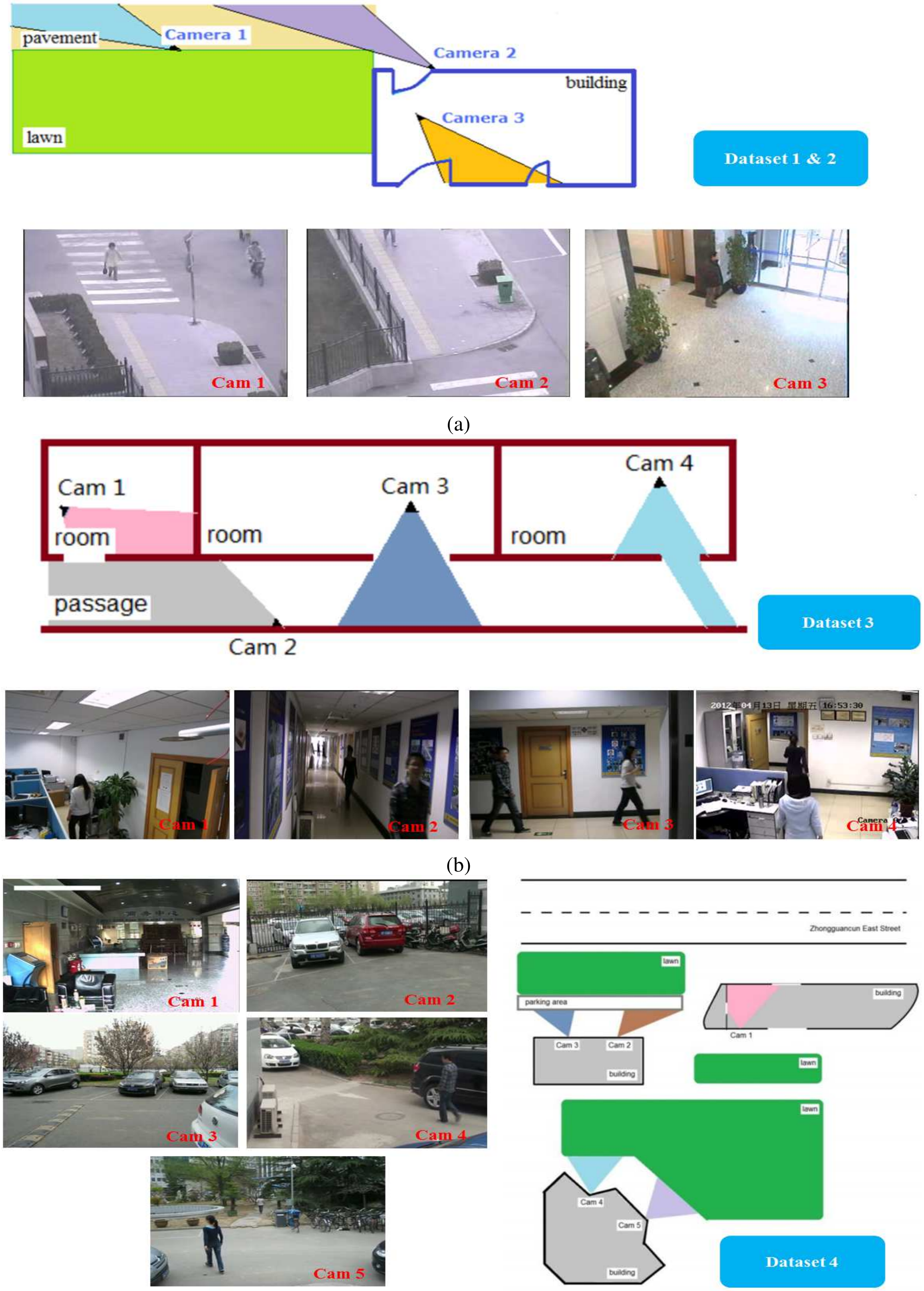
Work by [10], [11] proposed multi-target tracking by incorporating a pairwise exclusion effect between the targets at both *trajectory estimation* where any two trajectories should remain spatially independent as well as the *data association* step, in which each trajectory should be assigned at most one detection per frame. Kumar *et al.* [12] formulated multiple object tracking as a graph partitioning problem, in which the sum of weighted edges connecting graph nodes (target hypotheses) with the same label must be maximised. As a result, multiple trajectories indicating different targets are revealed. In more recent work [13], online multi-person tracking is formulated as decision making in Markov Decision Processes (MDPs), where the lifetime of an object is modelled with an MDP. Solera *et al.* [14] presented an online *divide and conquer* tracker for single static camera scenes, which partitions the targets assignment problem into local sub-problems and solves them by selectively choosing more reliable features.

B. Multi-View Multi-Person Tracking

In multi-camera multi-person tracking, due to the larger changes in targets' appearance, a more sophisticated appearance model should be designed to incorporate discriminative properties between true matches and wrong matches of target hypotheses in the non-overlapping cameras.

Kettnaker *et al.* [15] proposed the non-overlapping multi-camera tracking task as a Bayesian formalisation to recover the spatio-temporal trajectories of targets across cameras. Javed *et al.* [16] proposed a multi-camera tracking algorithm, which exploits the camera topology and path probabilities followed by people in the video scene. Dick *et al.* [17] introduced a real-time algorithm for tracking task that utilises a Markov model to track multiple targets in a disjoint camera views. In their work, people's motion patterns in both within and between fields of camera views are described by a stochastic transition matrix. Makris *et al.* [18] proposed the unsupervised learning base model of transition probabilities between non-overlapping camera views by extending the activity analysis between the cameras. Porikli *et al.* [19] presented an approach based on correlation matrix analysis and dynamic programming, where a distance metric and a model function are proposed to measure the inter-camera radiometric properties. The work of [20], [21] used the brightness transfer functions (BTFs) either online or offline between cameras to establish appearance similarities between targets in different cameras. Kuo *et al.* [22] proposed an online learned discriminative appearance affinity model for targets across multiple non-overlapping cameras, where Multiple Instance Learning (MIL) boosting algorithm is used to learn the discriminative appearance models for people.

Similar to our approach, there are some multi-camera tracking methods, which formulate target associations in both within and between cameras into a single global framework. Fleuret *et al.* [23] presented a framework to combine a generative model with dynamic programming to accurately track people in the presence of occlusions and lighting changes. Yu *et al.* [5] proposed a tracking-by-detection approach with non-negative discretisation for the data association problem, where multiple cues such as colour and face recognition information are utilised to establish target associations. The work of [24], [25] utilised a *min-cost flow* graph to associate different-view target hypotheses through their $3D$ reconstructed positions in a world coordinate space. In these methods, both temporal relationships as well as spatial correlations between targets are captured. However, these approaches are not applicable in a non-overlapping camera setting, where there is no information regarding the configuration of the cameras. The most recent similar work by [4] developed an approach that optimises the single-camera target tracking and inter-camera target tracking simultaneously in an equalised global graphical model. However for this purpose, an already-performed tracking (from



(a)

(b)

(c)

Fig. 1: An illustration of different non-overlapping multi-camera configurations for every sub-dataset of the *NLPR-MCT* dataset. These sub-datasets involve (a) *dataset 1-2*: outdoor scene, (b) *dataset 3*: indoor scene, and (c) *dataset 4*: outdoor scene, respectively. (Illustration from http://mct.idealtest.org/Datasets.html)

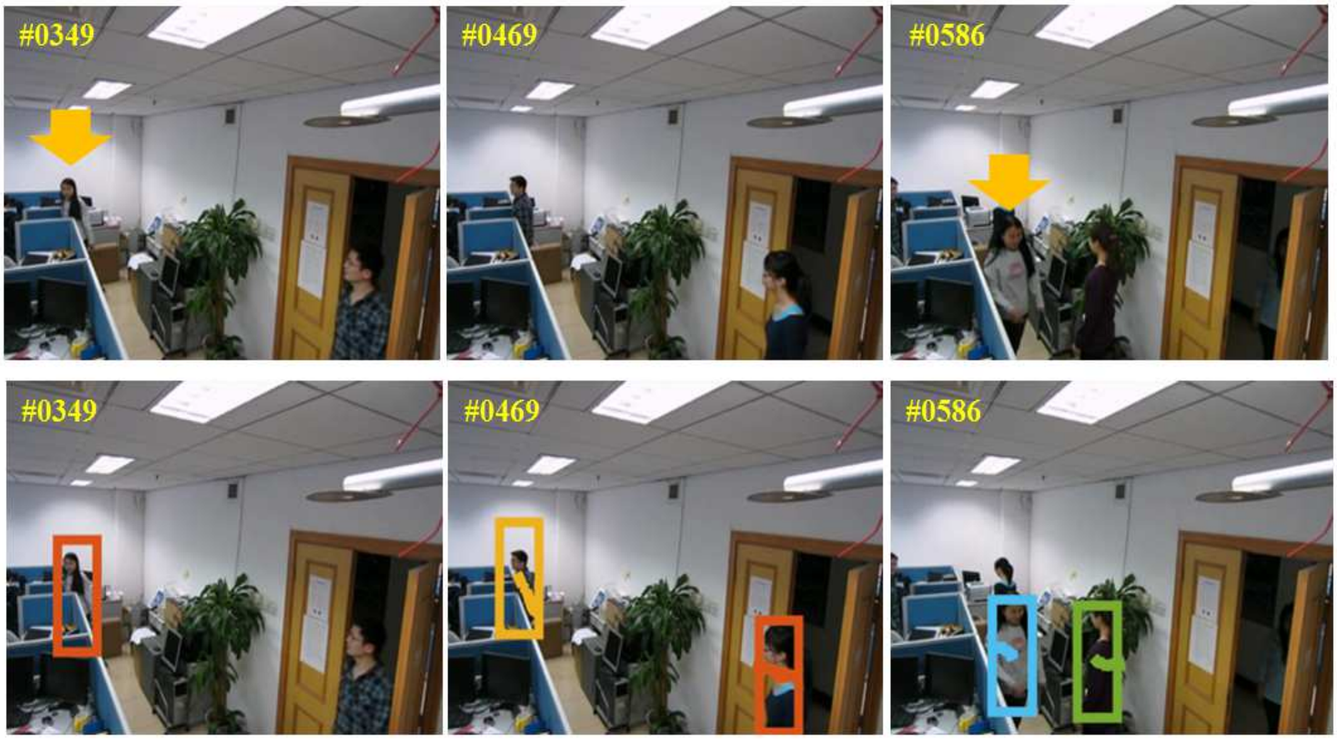


Fig. 2: Sample tracking results on the *NLPR-MCT* (*dataset 3*). Each colour corresponds to the unique person identity and identity switch (ID) occurs when the person leaves the scene and reappears with a delay. The highlighted walking person is assigned a new ID (shown with different colour) in the *second* row obtained by a state-of-the-art tracker [2].

[26]) is used to produce initial tracks as an input to the approach.

Tracking across non-overlapping camera views is closely related to the *re-identification* problem, where the correspondence between targets in cross-view cameras is usually governed by appearance features such as colour or behavioural biometrics. Wang *et al.* [27] proposed a novel framework to automatically learn a video ranking function for person re-identification. In order to handle inter-camera appearance variations, [28] presented an adaptive metric learning approach for a specific candidate set under the framework of transfer learning. Zhao *et al.* [29] proposed an approach based on unsupervised salience learning, where discriminative features are extracted without needing to identify target labels in the training process. Li *et al.* [30] introduced a Convolutional Neural Network (CNN) based framework – Deep Filter Pairing Neural Network (FPNN) – to jointly handle photometric and geometric transforms, misalignment and occlusion of targets.

C. Contributions

The contributions of this paper are briefly summarised below.

1) Multi-camera multi-person tracking is formulated as a multi-state graph optimisation problem (see Fig. 3) where all similarities, e.g. appearance similarity between any pair of target tracks (graph nodes) in both within and between cameras, are considered apart from their closeness in the temporal domain. Unlike previous approaches in multi-camera tracking task, which only consider association among targets between cameras, this approach solves data association in intra-camera and inter-cameras simultaneously in a single framework. A demonstration is available [1]. An iterative two-step scheme is proposed to solve the optimisation problem. As a result, multiple target tracks are revealed, where each track is smooth across video frames from different cameras. Moreover, we do not assume that the foreground targets appear within each local video segment of each camera and a selection matrix is presented to deal with occluded or fragmented targets.
2) We propose a learning framework of target-specific metrics, which helps to distinguish a specific target from other similar targets in a multi-camera network. We take advantage of both appearance and optical flow trajectories to provide more reliable affinity models to compare different target tracks (sequence of detections) across time and cameras.

III. Multi-State Multi-Camera Tracking Framework

Given a set of non-overlapping FOV cameras $\{C_1, \cdots, C_M\}$, we generate a set of initial (low level) target tracks $\mathcal{T} = \{\tau_1, \tau_2, \cdots, \tau_N\}$ within each camera as the input of our tracker. The i^{th} track τ_i is defined

[1] https://youtu.be/U3Qp8X9yP90

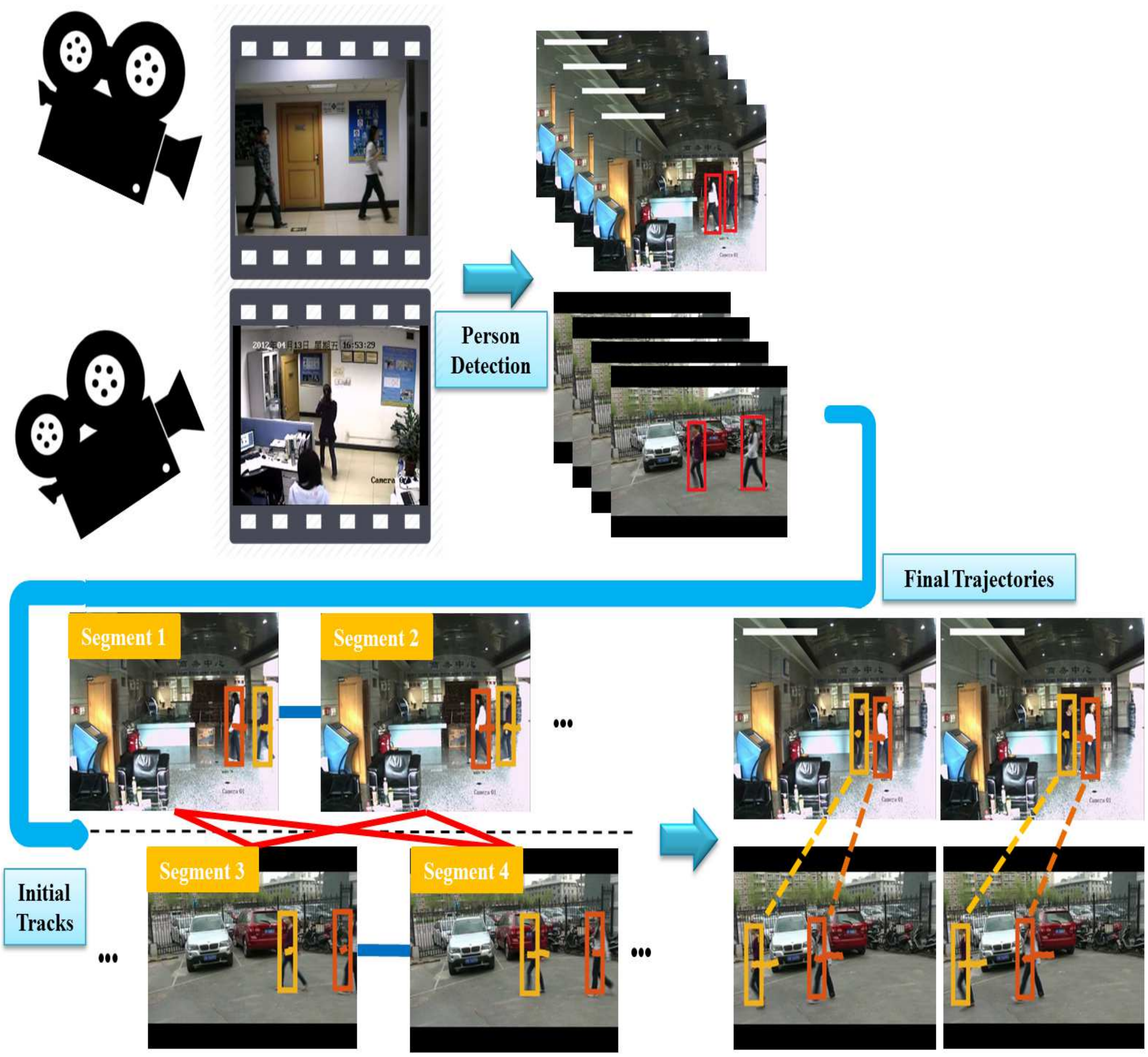


Fig. 3: Schematic illustration of the *MSMCT* algorithm. Starting from the initial target tracks within each camera view, obtained by a simple overlap criteria between detections (*red* bounding box) in consecutive frames, the video output for each camera is split into a number of different video segments and the association among all given target tracks in both within and between cameras are performed simultaneously in a single framework. As a result, a set of target trajectories, which are consistent among cameras, are obtained. Each colour represents the unique identity of a walking person and the bounding box tails represent the target track. *Blue* lines indicate the intra-camera similarity term, while *red* lines represent the inter-camera associations among the targets. Here, we only use two cameras for simplification.

as a set of detection hypotheses and is represented as $\tau_i = \left\{ O^i_{t^i_s:t^i_e}, 1 \le t^i_s \le t^i_e \le t \right\}$, which actually is a temporal sequence of the i^{th} track's detection responses O^i starting at time stamp t^i_s and ending at the end frame t^i_e. The generated tracks not only reduce the computational complexity but also allow incorporating motion information into the tracking framework. The notations used in this paper are presented in Table I.

In order to build initial target tracks within each camera, we use a *heuristic* approach. For each detection at time t, if it belongs to the set of non-collision detection responses of one target, it should correspond to the closest detection among the detection responses of the next video frame, while the second closest detection would correspond to a different target. Thus, for each detection, its *closest* and *second* closest detection for the next frame are considered and the ratio of their distance[2] is calculated. A ratio smaller than a threshold[3] means that both detections belong to the same target. To increase the reliability of these initial tracks, their lengths are bounded between $[6, 12]$ frames.

Given a set of target tracks in each camera, we propose a multi-state graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ with a set of vertices $\mathcal{V}$ and edges $\mathcal{E}$ to model the relationship among different target tracks in a set of videos. In this graph structure, the target track

[2] Position and object size features are used.

[3] The threshold is empirically set to 0.3.

TABLE I: Notations used in this paper.

Notation symbol	Explanation
M	Number of cameras
K	Number of target identities
N	Number of initial target tracks
O	Set of target object hypotheses (e.g. detection responses)
O_t^i	Hypothesis of target object i at frame t
C_i	The i^{th} camera's field of view
$Conf^i$	The average confidence score of the i^{th} track detections
p_t^j	Position of target object j at frame t
σ	The standard deviation for the target size
Δ_f	Frame gap
Δt	Time interval
$\mathcal{T}$	Set of all targets tracks
t_s^i	Starting time index for the i^{th} target track
t_e^i	Ending time index for the i^{th} target track
i	A node of spatio-temporal graph, *i.e.* a track index
τ_i	The i^{th} target track
e_{ij}	Transition edge between the track pair (τ_i, τ_j)
$\phi(\cdot)$	*Unary* cost for each graph node
$\psi(\cdot,\cdot)$	*Pairwise* energy term
ψ_{intra}	Intra-camera tracks affinity term
ψ_{inter}	Inter-camera tracks affinity term
$\Phi\left(x^{(k)}, x^{(j)}\right)$	Exclusion effect between pairwise targets' states ($x^{(k)}$ and $x^{(j)}$)
θ	The parameter adjusts the number of track pairs included in each positive bag
(t_1, t_2)	The indices of frames closest in time for the two tracks
$Tr(\tau_i, \tau_j)$	All common point trajectories overlapping track pair (τ_i, τ_j)
tr_o	A sequence of space-time point trajectory (the o^{th} point trajectory)
D_i^t	Detection response at frame t for the track τ_i
loc	Euclidean (relative) distance function
f_i	The feature vector belongs to the i^{th} target track
y^p	Positive instance
y^n	Negative instance
h	Distance function
B_i^+	Positive bag for the i^{th} track
B_i^-	Negative bag for the i^{th} track
γ	Normalisation parameter for the appearance-based affinity model
P_j^{head}	The position of the *head* of track j
P_i^{tail}	The position of the *tail* of track i
v_j^{head}	The velocity evaluated of the head of track j
v_i^{tail}	The velocity evaluated of the tail of track i
P_a^{exit}	The disappearing point of exit area in Camera C_a
P_b^{enter}	The disappearing point of enter area in Camera C_b
$x^{(i)}$	The spatio-temporal state series of the i^{th} target
H_i^b	Similar track set $\left\{\tau_j^b\right\}$ w.r.t τ_i^a in camera C_b
W	Target track metrics
z_i^t	Feature vector of a detection response in track τ_i at frame t
g_i	Probe detection responses (detection with the larger confidence score) of the track index i
d_{ij}	Average relative distance between detections of track pair (τ_i, τ_j)
T	Number of collected training features from the tracklets

within each video segment of one camera is a graph node, and the target candidates are the states that this node can take. Therefore, each graph node can take a state x_m from a configuration set $\left\{x_1, \cdots, x_{|\mathcal{V}|}\right\}$. These states indicate spatio-temporal paths of the targets during tracking. We integrate multiple state selection in the graph that builds upon an energy minimisation technique, which outputs final trajectories in a cross-view camera network, simultaneously. Each output trajectory in a multi-camera network is not only expected to belong to the same target within an individual camera view, but also consistent among different cameras.

Graph *edges* $\mathcal{E}$ are also vital components for our graph design. Although the graph nodes (initial tracks) from different cameras are fully connected in the graph structure, the topological connection between any two cameras should allow the transition edge e_{ij} between the track pairs (τ_i, τ_j) by considering a panoramic view. Moreover, there should be a maximum frame gap between any track pairs in cross-cameras, in which if the frame interval between two tracks is greater than a threshold Δ_f, they are unlikely to be connected:[4]

$$\mathcal{E} = \left\{ (e_{ij})_{intra} \right\} \cup \left\{ (e_{ij})_{inter} \right\}, \quad s.t. \quad t_s^j - t_e^i < \frac{\Delta_f}{fps} \quad (1)$$

These transition edges include both inter-camera edges $(e_{ij})_{inter}$ and intra-camera edges $(e_{ij})_{intra}$.

The edge *costs* represent the similarities among the graph nodes (target tracks). It should be noted that using the same similarity measure, due to the larger variations in lighting conditions and viewpoints in inter-camera tracking compared with intra-camera tracking, the similarity score between two tracks in different cameras is usually lower than the score of tracks within the same camera. Therefore, the normalisations of these two similarity metrics is not straightforward. In this paper, we consider two sets of edge costs involving intra-camera and inter-camera affinity measures.

In the proposed multi-state graph optimisation problem, the aim is to find the most probable states for each graph node (target track), which is equivalent to minimising the energy function: $\hat{x} = arg \min_x E(x)$ and is defined as:

$$E_{MSMCT}(x) = \sum_{m \in \nu} \phi(x_m) + \sum_{(m,l) \in \varepsilon} \psi(x_m, x_l) \quad (2)$$

where $\phi(x_m)$ is the *unary* cost for the node state x_m and $\psi(x_m, x_l)$ is a *pairwise* energy term assigned to a mutual edge $(m, l) \in \varepsilon$ between the track indices m and l.

Suppose, there are K different target identities (labels) and $x^{(i)}$ representing the spatio-temporal state series of the i^{th} target in a cross-camera network. Considering the mutual exclusion $\Phi\left(x^{(k)}, x^{(j)}\right)$ between any target pairs[5], which favours different tracks belonging to different targets to be separated, we can extend Eq. (2) to formulate the multiple states' graph optimisation as:

$$E_{MSMCT} = \sum_{k=1}^{K} \left[\sum_{m \in \nu} \phi\left(x_m^{(k)}\right) + \sum_{(m,l) \in \varepsilon} \psi\left(x_m^{(k)}, x_l^{(k)}\right) \right] + \sum_{(k,j)} \sum_{m \in \nu} \Phi\left(x_m^{(k)}, x_m^{(j)}\right) \quad (3)$$

In fact, the pairwise cost ψ measures the compatibility between any two tracks that belong to the same target. This can be decomposed into an *intra-camera* track affinity ψ_{intra} and an *inter-camera* affinity term ψ_{inter}.

In particular, for the intra-camera pairwise affinity metric between the tracks ψ_{intra}, we break down each camera's image sequences into the different segments and generate a pool of initial tracks within each segment. Then, we utilise *optical flow* to define a likelihood of matching tracks from different video segments of each camera view.

To model the pairwise affinity of the tracks in disjoint camera views ψ_{inter}, the *target-specific* appearance-based model is presented, which can help to distinguish the features belonging to the same target from those corresponding to different targets. The value K can be an overestimation of the actual number of targets. We now explain the components of the *MSMCT* energy in more detail.

A. Intra-Camera Affinity of Tracks

As far as the association between the target tracks within a camera is concerned, the spatio-temporal path for targets should be smooth and continuous. For example, when a target leaves the scene and then comes back to the FOV of the camera with a delay, it is difficult to determine whether the tracks (partial trajectories) obtained by the tracker correspond to the unique target or should be considered as fragmented tracks, which belong to two different targets. Thus, a robust pairwise affinity measure is needed to handle this difficulty. Fragkiadaki *et al.* [31] proposed a method to associate multiple targets using optical flow trajectories. However, their model is sophisticated due to the joint inference on both flow trajectories and target detection responses. Here, we utilise *dense point* trajectories to encode the relative motion pattern between two tracks in a temporal distance [32]. The aim is to find the common dense point trajectories overlapping between the set of bounding boxes of the detections within the tracks $Tr(\tau_i, \tau_j)$ to build their mutual affinities ψ_{intra} as shown in Fig. 4. Thus, the smooth affinity between their frames closest in time (t_1, t_2) is computed as:

$$\psi_{intra}(x_i, x_j) = exp\left(- \sum_{o \in Tr(\tau_i, \tau_j)} \frac{\left| loc\left(tr_o, D_i^{t_1}\right) - loc\left(tr_o, D_j^{t_2}\right) \right|^2}{\sigma_l^2} \right) \quad (4)$$

where D_i^t is the detection response at frame t for the track τ_i, $loc\left(tr_o, D_i^t\right)$ denotes the *Euclidean distance* between the dense (point) trajectory tr_o and the centre of detection bounding box D_i at frame t. The parameter σ^2 adjusts the sensitivity of the affinity value to the abrupt target size change[6]. The affinity score for the two tracks is higher when their common dense trajectories are consistent with their detection responses. $\psi_{intra}(x_i, x_j) = 0$, if there are no common dense point trajectories between two related tracks. Thus, the proposed affinity considers the long-term similarity between the target candidates, which are not necessarily in consecutive frames.

B. Inter-Camera Affinity of Tracks

In this section, we first present the target-specific metric learning. Then, the appearance-based affinity model between tracks in a cross-camera network is discussed, which is used as inter-camera affinity ψ_{inter} of the tracks in the multi-state graph optimisation framework.

We seek to learn distinguishable target-specific metrics while keeping the computational complexity low. Accordingly, the learning models should distinguish any two tracks from different cameras belonging to the same target from those of other targets. The framework involves three main steps:

[4] This dataset-specific threshold is defined by computing the maximum cross-camera transition frame gap of the targets and is set to 1300 frames and the videos frame rate is 20Hz.

[5] Any target track pairs from the set of target spatio-temporal series $\left\{x^{(1)}, \cdots, x^{(K)}\right\}$

[6] The variance is set to $\sigma^2 = 4$, experimentally.

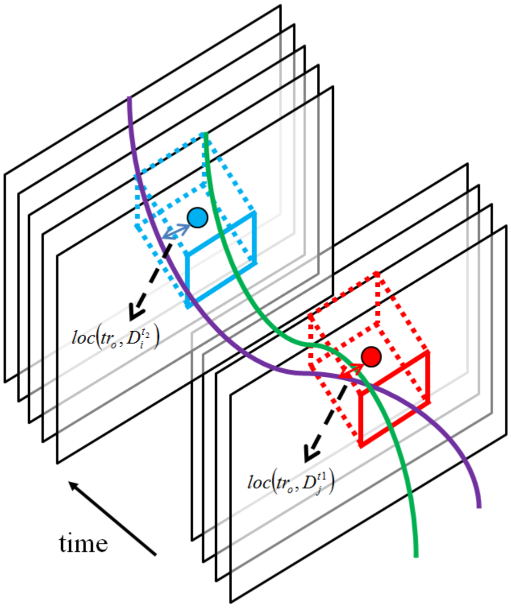


Fig. 4: Illustration of the intra-camera affinity model between the target tracks. The detection responses within the target tracks are linked using point trajectories.

(1) collecting training samples, (2) learning the target-specific model, and (3) contracting the appearance affinity between target tracks in different cameras.

Given the training samples for the tracks from any pair of non-overlapping cameras views $\{f_i\}_{i=1}^{T}$, where $f_i \in \Re^{N_d}$ is the feature vector belonging to the i^{th} track[7] and N_d is the total number of feature dimensions, learning the target-specific model is formulated as a *distance metric* function, which can enhance discrimination between feature vectors by ensuring that distances are smaller when features are extracted within the pair of tracks of the same person from multi-views domain, and larger otherwise.

Inspired by Multiple Instance Learning (MIL) [33], where each training sample is a *bag* of instances, we devise a similar framework for the cross-view track similarities. In particular, we define a *positive* instance as a difference vector between a pair of tracks belonging to the same target and a *negative* instance as a difference vector computed from any pair of tracks belonging to different targets:

$$\begin{aligned} y^p &= \left|f_i^a - f_i^b\right| \\ y^n &= \left|f_i^a - f_j^b\right| \end{aligned} \tag{5}$$

where f_i^a and f_i^b are the feature vectors of the *same* track i, captured in cameras C_a and C_b, respectively, while f_j^b is the feature vector obtained by the track j in camera C_b. Given these difference vectors, the objective is to learn a distance function h based on the mutual distance comparison of the cross-view target tracks pairs:[8]

$$h\left(\left|f_i^a - f_i^b\right|\right) = w^T \left|f_i^a - f_i^b\right| \tag{6}$$

which favours the most similar cross-view track pair of the same track i, (positive instances) over those of two different persons (negative instances) as in Eq. (7). By listing all possible instances between cross-view target track sets for the track i, we form a positive bag $B_i^+ = \{y^p\}$ and a negative bag $B_i^- = \{y^n\}$, respectively.

$$\max_{y^p \in B_i^+} h\left(y^p\right) > h\left(y^n\right), \quad s.t. \quad \forall y^n \in B_i^- \tag{7}$$

Since in any two non-overlapping cameras, one person cannot appear at the same time in both cameras' FOV, the negative samples are collected by considering pairs of tracks in two cameras, which overlap in time. However, due to the absence of supervised information, it is not straightforward to determine whether two tracks across two cameras represent the same person or not. For this scenario, instead of labelling individual samples, we impose spatio-temporal constraints on the cross-view tracks. For a certain track τ_i^a and its corresponding feature vector f_i^a in camera C_a, a set of its *similar* track pairs $H_i^b = \left\{\tau_j^b\right\}$ in camera C_b is considered to form the positive '*bag*'.

Similar to [4], we annotate the *enter/exit* areas of individual cameras to relocate any target, which disappears from the FOV of one camera and reappears with a delay in the enter area of another camera as in Fig. 5. In addition, in order to link the cameras, we label a *fading* point for each area, where the related target appears or disappears. Then, the motion similarity between any two tracks (τ_i, τ_j) is computed, which is based on the relative distances of cross-view tracks to the *fading* point assuming a linear motion model. In particular, considering positions of the two tracks P_j^{head} and P_i^{tail} with the time interval Δt between them, the relative distances with respect to the *fading* point are computed as:

$$\Delta P_i = \min_{t \in [1, \Delta t]} \left\|\left(P_i^{tail} + v_i^{tail}\Delta t\right) - P_a^{exit}\right\|_2 \tag{8}$$

$$\Delta P_j = \min_{t \in [1, \Delta t]} \left\|\left(P_j^{head} - v_j^{head}\Delta t\right) - P_b^{enter}\right\|_2 \tag{9}$$

where v_i^{tail} and v_j^{head} are the velocities evaluated from the tail of track i and the head of track j, respectively. P_a^{exit} and P_b^{enter} are the positions of the *fading* points for exit area and enter area in cameras C_a and C_b, respectively. Finally, we choose those cross-view tracks to constitute the positive bag with a high probability of motion similarity as:

$$\tau_j^b \in H_i^b \quad if \quad exp\left(-\lambda\left(\Delta P_i + \Delta P_j\right)\right) > \theta \tag{10}$$

where λ is set to 0.005 in the experiments and the threshold θ is chosen experimentally to control the number of track pairs included in each positive bag.

In order to solve an optimisation problem of Eq. (6), which satisfies a constraint in Eq. (7), we formulate it into the *max-margin* framework as in [34] and relax this problem into a non-

[7] For each target track, we use a *HSV* colour histogram. The histogram is calculated for all detection responses of one track and the median appearance of the detections is considered as the appearance representation of the track.

[8] w is the target track metric for the distance function h.

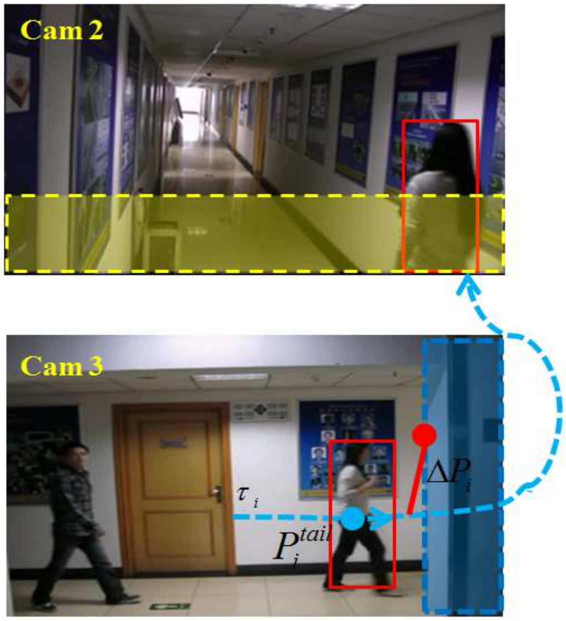


Fig. 5: Demonstration of the relative distance of the target tracks in the disjoint cameras views. The *blue* and *yellow* polygons denote the exit and enter area used in the experiments, respectively. The *red* lines demonstrates the relative distances of the tracks with respect to the fading point.

constrained primal problem, which can be solved efficiently by the linear conjugate gradient approach:

$$w^* = arg\min_{w,\xi} \frac{1}{2}\left\|w\right\|^2 + C \sum_{\left\{y_k \in B_i^-\right\}} l\left(0, 1 - w\left(Y_i^T - y_k^T\right)\right)^2 \tag{11}$$

where l denotes the *hinge loss* function and ξ is the slack vector. Each column of Y_i is equivalent to one $y^p \in B_i^+$.

As a result, we learn the target-specific models $W = \{w_i\}_{i=1}^T$ for all tracks in the cross-camera network. Eventually, to calculate the appearance affinity models between the tracks ψ_{inter} in the cross-camera network, we compute the mean values of the relative distances between tracks:

$$\begin{gathered} y_{ij}^t = \left|z_i^t - g_j\right|, \quad y_{ji}^{t'} = \left|z_j^{t'} - g_i\right| \\ d_{ij} = mean\left(\left\|W_i^T y_{ij}^t\right\|^2\right), \quad d_{ji}^{t'} = mean\left(\left\|W_j^T y_{ji}^{t'}\right\|^2\right) \end{gathered} \tag{12}$$

where z_i^t represents the feature vector[9] of a detection response in track τ_i at frame t, $z_j^{t'}$ denotes the feature vector of a detection response in track τ_j at frame t', and g_i, g_j are probe detection responses (strongest detection responses form either tracks with higher confidence scores). Hence, the appearance affinity model is defined as:

$$\psi_{inter}\left(x_i, x_j\right) = \gamma\left(d_{ij} d_{ji}\right)^{-1} \tag{13}$$

where γ is the normalisation parameter.

C. Unary Cost

The *unary* cost $\phi\left(x_i\right)$ for the nodes in the proposed graph can be defined using the average confidence score $Conf^i$ of the detections for the related target track and its corresponding appearance metric obtained from Eq. (11) as:

$$\phi\left(x_i\right) = \left|W_i^T Conf^i\right| \tag{14}$$

If only ground truth is used for the target hypotheses, we set the confidence score to one, otherwise the confidence score weights the unary costs for the graph nodes indicating the presence of foreground targets.

D. Target Exclusion Cost

We add a large penalty to those target tracks, which overlap in time, since they will not belong to the same target. It means that if two presumed target positions are very close to each other, a large penalty will be added to the cost function unless one of them is an outlier.

$$\Phi\left(x^{(k)}, x^{(j)}\right) = w_E \cdot \frac{\left|x^{(k)} \cap x^{(j)}\right|}{\left|x^{(k)} \cup x^{(j)}\right|} \tag{15}$$

where w_E is the exclusion cost parameter.

IV. Occlusion Handling

The output of the optimisation in Eq. (3) are target trajectory hypotheses corresponding to a set of states participating in the *MSMCT* solution. However, a track of a person may be occluded and may not necessarily be present in all video segments within each camera view. In order to avoid selecting irrelevant states in a track of a person, we present a *selection* matrix S for each camera view. Assume $S \in \Re^{M \times K}$ represents the selection matrix for a cross-camera network, where each entry s_{mk} determines whether camera view m contains target k. If it does, then we set $s_{mk} = 1$, otherwise, we set it to zero. This approach will ensure the output target tracks will be clear of outliers. Considering the selection matrix in the graph optimisation, we can reformulate the energy function in Eq. (3) as:

$$\begin{aligned} min \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i,j \in C_m} s_{mk} \left[\phi\left(x_{m,i}^{(k)}\right) + \psi_{intra}\left(x_{m,i}^{(k)}, x_{m,j}^{(k)}\right)\right] \\ + \sum_{k=1}^{K} \sum_{n \neq m=1}^{M} \sum_{i \in C_m, s \in C_n} s_{mk} s_{nk} \phi_B\left(x_{m,i}^{(k)}, x_{n,s}^{(k)}\right) \\ + \sum_{k \neq l=1}^{K} \sum_{m=1}^{M} \sum_{i \in C_m} s_{mk} s_{ml} \Phi\left(x_{m,i}^{(k)}, x_{m,i}^{(l)}\right) \end{aligned} \tag{16}$$

A. Solving the Optimisation Problem

We seek to automatically discover the most similar cross-view video track pair for multiple persons simultaneously through the optimisation. Thus, we solve Eq. (16) by optimising this problem *iteratively* between two sets of variables: the selection matrix S and the set of targets states $\left\{x^{(1)}, \cdots, x^{(K)}\right\}$.

[9] The *HSV* colour histogram is used.

a) Target States Series: We fix the selection matrix S to optimise the candidate spatio-temporal series $\{x^{(1)}, \cdots, x^{(K)}\}$. This energy minimisation problem can be solved by Tree Re-Weighted Message Passing (**TRW-S**) [35].

b) Selection Matrix: In the selection step, using the updated target states from the previous step, we update the selection matrix based on the *unary* cost of the tracks, in which if the corresponding target state has lower unary scores, it is expected to be occluded in a corresponding video segment and we assign a pre-defined value ϵ. For initialisation, all values in the selection matrix S are set to one. During this iterative optimisation, multiple cross-view similar tracks, each belongs to different target with less noises and occlusion will be extracted. The iterations can be stopped when the indicator matrix does not change or upon reaching a predefined maximum number of iterations. In the experiments, the optimisation algorithm terminates in *4-5* iterations for the multi-camera tracking task.

V. Experiments and Results

In this paper, the performance of the proposed tracking framework is evaluated with several state-of-the-art multi-camera trackers from the Multi-Camera Object Tracking *MCT* Challenge [36] such as *hfutdspmct* [36], *USC-Vision* [37], *CRIPAC-MCT* [38] and an equalised global graphical model-based approach proposed by Cao *et al.* [4]. In our experiments, we utilise the new introduced benchmark, namely the *NLPR-MCT* dataset [1], as well as new evaluation criteria for the multi-camera tracking task, which are specialised for the non-overlapping multi-camera settings. For a fair comparison, all results are computed using the same provided annotations.

The MCT Challenge consists of *three* different experiments to yield better evaluation of a tracker's performance. The details of these three experiments are:

- **Exp. 1:** A tracker is executed on all videos in the *NLPR-MCT* dataset by the provided ground truth annotations obtained by the single-camera object tracking results. The only objective here is to find the data association across different cameras.
- **Exp. 2:** A tracker is executed on all videos with the provided ground truth detections. Here, both data association (within and between cameras) are not available.
- **Exp. 3:** This experiment runs on all videos in the *NLPR-MCT* dataset without using any annotations.

Since the single-camera data association as well as the multi-camera data association among targets are formulated in an united framework in our approach, *experiment* 1 is not applicable here and we use *experiments* 2 and 3 to evaluate the suggested tracker's performance. We use a deformable part based model [39] to obtain the detection hypotheses O in each video frame for *experiment* 3. In addition, to verify the effectiveness of different components of the proposed approach such as the within and between camera appearance affinity models, two different experiments are devised to compare the performance of the proposed tracker with different *constraints*.

A. Performance Measure

Quantitative evaluation of multiple target tracking across multiple cameras is a difficult task due to the different evaluation strategies between a single-camera tracking and multi-camera tracking task. In single-camera tracking, the *CLEAR MOT* metrics [40] and metrics from [22], [41] are two commonly used performance measures, which consider different factors for the tracker's assessment. For example, the *MOTP* metric in *CLEAR MOT* measures the tightness of the tracking results and ground truth, while some other trajectory based metrics such as *MT* [22] measure the percentage of mostly tracked targets, which their trajectories are successfully tracked for more than 80% of their ground truth time span.

However, these metrics may not be appropriate measures to evaluate the performance of a tracker across multiple cameras, where there are different types of ID switches among closely moving targets. Moreover, the provided ground truths for the multi-camera tracking are much less than in a single-camera tracking task in *NLPR-MCT* datasets. For this reason, a suitable metric is needed to address this specific problem. In this paper, we use a new evaluation criterion, namely *MCTA*, to take these precedents into account. In particular, this evaluation metric formulates ID-switches in both intra-camera and inter-camera domains in a single metric and assigns an equal importance to each one as:

$$\begin{aligned} MCTA = \left(\frac{2 \times Precision \times Recall}{Precision + Recall}\right) \cdot \\ \left(1 - \frac{\sum_t mme_t^s}{\sum_t tp_t^s}\right) \times \left(1 - \frac{\sum_t mme_t^c}{\sum_t tp_t^c}\right) \end{aligned} \tag{17}$$

The *MCTA* metric is bounded between $[0, 1]$ and consists of three components: (1) detection accuracy and (2, 3) two other modified metrics of *CLEAR MOT* [40] representing the single-camera and cross-view tracker's abilities, respectively. Each of them corresponds to an individual bracket in Eq. (17).

The first term accounts for the detection ability of the tracker and is composed of two more terms:

- **REC** (↑) :[10] Recall, the ratio of the correctly matched detections over the total number of detections in the ground truth.
- **PRE** (↑) : Precision, the ratio of the correctly matched detections over the total number of output detections.

and are formulated as:

$$\begin{aligned} Precision &= \left(1 - \frac{\sum_t fp_t}{\sum_t r_t}\right) \\ Recall &= \left(1 - \frac{\sum_t m_t}{\sum_t g_t}\right) \end{aligned} \tag{18}$$

where m_t, fp_t, g_t and r_t are *missed targets*, *false positive*, *ground truth* and the number of output *target hypotheses*, respectively at frame t.

The *second* and *third* components of Eq. (17) represent the single-camera and cross-camera tracking capabilities based on the target mismatches, respectively. In particular, mme_t^s denotes for the number of mismatches in a single-camera, while

[10] Here, the symbol ↑ means that higher scores indicate better results.

mme_t^c represents the number of ID-switches (mismatches) in cross-camera tracking. Besides, tp_t^s and tp_t^c also denote the number of true positives (ground truth) for time t in a single-camera and across different cameras, respectively.

B. Datasets

The performance of the proposed approach is assessed on the *NLPR-MCT* dataset as a recent non-overlapping multi-camera tracking data source. Our key motivation for using this dataset is that while other datasets (usually designed for single-camera tracking [42], [43]) exist for evaluating tracking methods, this new multi-person dataset is well suited in the context of non-overlapping multi-cameras and simplifies tracker evaluation. The *NLPR-MCT* dataset consists of *four* subsets. Every sub-dataset involves $3-5$ cameras and has different camera configurations based on the number of people (ranging from 14 to 255). The many illumination changes, mutual occlusions and interactions among people make this dataset very challenging. These subsets are briefly summarised below.

- **Datasets 1 & 2:** In both datasets, there are *three* synchronous videos from non-overlapping cameras with a frame rate of 20Hz and a resolution of 320×240 pixels. These videos are captured by two outdoor cameras and one indoor camera. There are different challenges such as illumination changes between outdoor and indoor scenarios and severe occlusion, especially in *dataset* 2. The total duration of both datasets is 20 min. The number of persons in *dataset* 1 and *dataset* 2 is 235 and 255, respectively.
- **Dataset 3:** In contrast to the other datasets, which are nearly 20min long, *dataset* 3 lasts about 4min with a frame rate of 25Hz and contains *four* synchronous videos recorded from non-overlapping *indoor* cameras. Moreover, several visually similar targets move closely together, causing an ambiguity problem in recovering target trajectories.
- **Dataset 4:** This dataset consists of *five* synchronous videos from five non-overlapping cameras. One is indoors, while the other four cameras are mounted outdoors. There are also serious illumination variations across the cameras.

C. Evaluation of the Proposed Target Affinity Models

In order to prove the effectiveness of the appearance affinity models used in the proposed framework, two different experiments are conducted to compare the performance of the tracking system with different constraints:

- **Constraint 1:** System without the learned intra-camera affinity model of the tracks.
- **Constraint 2:** System without the learned inter-camera affinity model of the tracks.

In the first experiment (*Constraint* 1), the proposed intra-camera similarity metric is replaced by the appearance affinity computed by the colour *histogram intersection* between the two tracks. For this purpose, we use the colour histogram [44] for the appearance representations of the targets. The histogram is calculated for all detections of one track and the mean appearance of the detections is chosen as the appearance representation of the corresponding target track. The appearance affinity is computed by histogram intersection between the two appearance descriptors of the tracks.

The tracking results of the proposed tracker *without* considering the inter-camera affinity model (*Exp. 2*) are shown in Tables VI and VII. In this experiment, the proposed target-specific discriminative affinity model is replaced by the similarity measurement based on the major colour spectrum histogram representation (MCSHR) [45] of the cross-view tracks.

D. Sensitivity Test

The proposed tracker's sensitivity under different parameter settings is examined for the different sequences. Fig. 6 shows the sensitivity test results for the number of targets mismatches for increasing the σ in *NLPR-MCT* dataset. Since the size of the tracked target will be changed from frame to frame, the standard deviation σ for the target size noises must be adjusted to handle the size variations. As shown in the Fig. 6 (a,b), with increasing target size deviation σ, the number of mismatches in a single-camera will increase due to the larger number of pairwise relationships between incorrect target tracks in the defined tracks affinity model. The figures for the number of mismatches in cross-camera view remained relatively stable and do not change sharply (see Fig. 6 (c,d)). The proposed tracker's performance *MCTA* dropped slightly with increasing the standard deviation σ (see Fig. 7). However, due to the large number of true positive targets in a single-camera tp^s as denominator of Eq. (17), these changes do not affect much the proposed tracker's performance. In our experiments, the standard deviation σ has been set to 2 (optimised for MCTA and for a target height of 70 pixels).

For the frame interval Δ_f, we estimate a minimum of the waiting time $\frac{\Delta_f}{fps}$ for the *NLPR-MCT* dataset and use it as default: 60s (1200 for $20fps$ in *datasets* 1 *and* 2, 1500 for $25fps$ in *datasets* 3 *and* 4, respectively). Then, we changed the default in a range $(60s \sim 70s)$, and test the tracker's accuracy (see Fig. 8). The parameter with the best performance is adopted as our waiting time.

The parameter θ in Eq. (10) is empirically set to 0.3 to maintain a moderate number of track pairs included in each positive bag and save the data-association run-time cost. We have examined the values of the θ in a range $(0.1 \sim 0.9)$ and observe that the accuracy performance is not influenced much by the this parameter. All other parameters including e.g. γ are determined experimentally (optimised for MCTA). A single parameter set is used for all datasets. We discovered that the proximity information of inter-target mutual occlusion in a scene has a great influence on the best choice of parameters such as w_E.

VI. Computational Complexity

The run time of the proposed method largely depends on the number of targets (graph nodes) in the disjoint camera views.

TABLE II: Performance comparison between the proposed tracker and other state-of-the-art methods in *experiment 2*. We use the ground truth of object detection as input for all datasets.

Datasets	Method	mme^s	mme^c	MCTA
Dataset 1	*hfutdspmct* [36]	77	84	0.74
	USC-Vision [37]	63	35	0.88
	CRIPAC-MCT [38]	135	103	0.69
	Equalised Model [4]	66	49	0.85
	MSMCT	68	42	0.86
Dataset 2	*hfutdspmct* [36]	109	140	0.65
	USC-Vision [37]	61	59	0.83
	CRIPAC-MCT [38]	230	153	0.62
	Equalised Model [4]	93	107	0.73
	MSMCT	63	67	0.81
Dataset 3	*hfutdspmct* [36]	105	121	0.20
	USC-Vision [37]	93	111	0.24
	CRIPAC-MCT [38]	147	139	0.08
	Equalised Model [4]	51	80	0.47
	MSMCT	45	64	0.58
Dataset 4	*hfutdspmct* [36]	97	188	0.26
	USC-Vision [37]	70	141	0.43
	CRIPAC-MCT [38]	140	209	0.18
	Equalised Model [4]	128	159	0.37
	MSMCT	61	132	0.49

TABLE IV: Performance comparison using the average results of the trackers over four sub-datasets in *experiment 2*.

	Method	mme^s	mme^c	MCTA
Average	*hfutdspmct* [36]	97	133	0.46
	USC-Vision [37]	71	86	0.59
	CRIPAC-MCT [38]	163	151	0.39
	Equalised Model [4]	84	98	0.60
	MSMCT	**59**	**76**	**0.68**

TABLE VI: Quantitative evaluation results for the main system parts in *Experiment 2*. The constraint 1 is a system without the learned intra-camera affinity model of the tracks while the constrains 2 refers to a system without the learned inter-camera affinity model.

Datasets	Brief Description	mme^s	mme^c	MCTA
Dataset 1	*Constraint 1*	79	57	0.81
	Constraint 2	93	89	0.72
	$OverallSystem$	68	42	0.86
Dataset 2	*Constraint 1*	89	96	0.76
	Constraint 2	94	103	0.74
	$OverallSystem$	63	67	0.81
Dataset 3	*Constraint 1*	76	70	0.53
	Constraint 2	81	76	0.49
	$OverallSystem$	45	64	0.58
Dataset 4	*Constraint 1*	69	137	0.45
	Constraint 2	73	145	0.42
	$OverallSystem$	61	132	0.49

TABLE III: Performance comparison between the proposed tracker and other state-of-the-art methods in *experiment 3*. We use the deformable part based model [39] for target object detection in all datasets. *NA* signifies results that are not available.

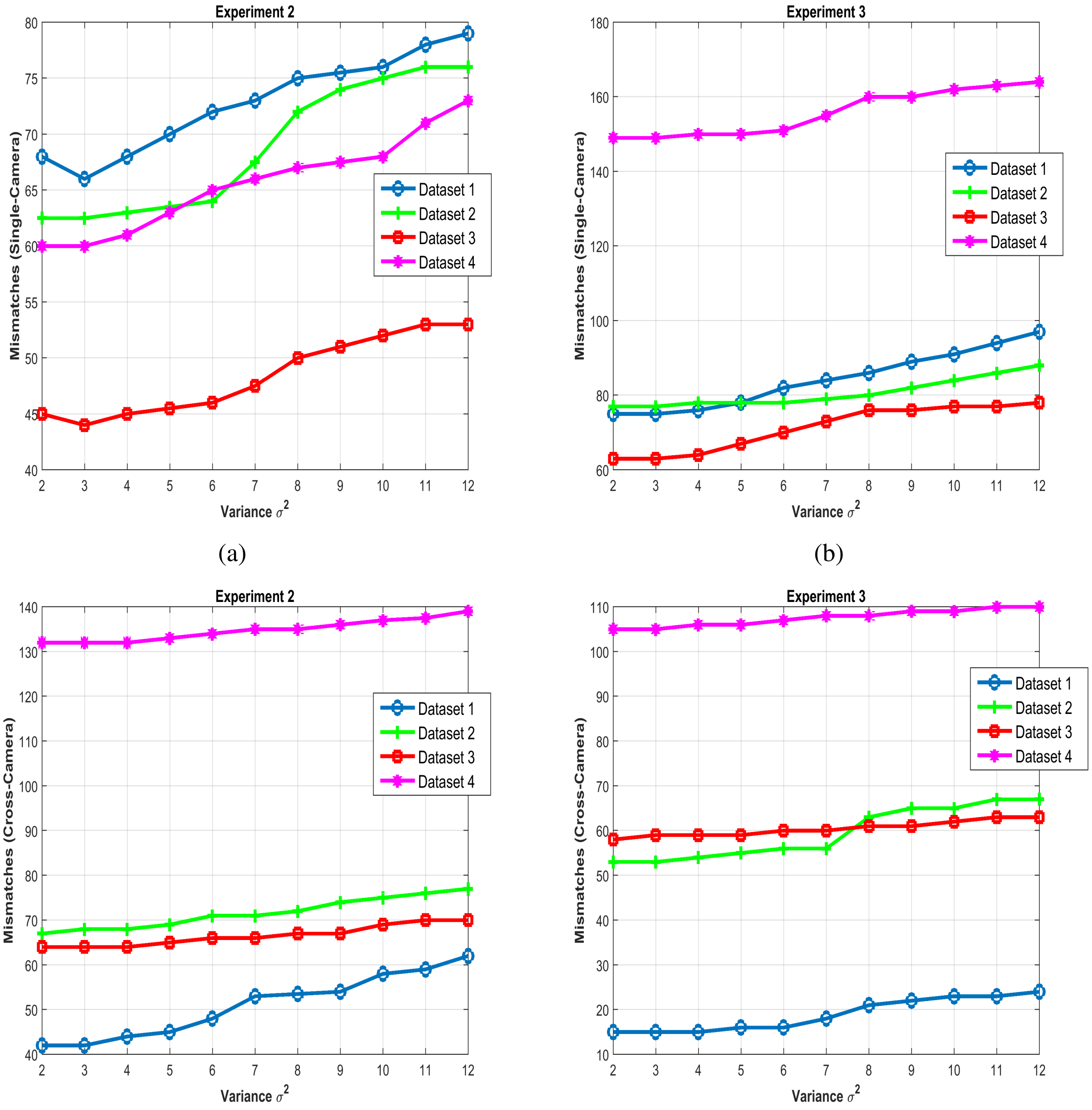
Dataset	Method	mme^s	mme^c	MCTA
Dataset 1	*hfutdspmct* [36]	1903	113	0.28
	USC-Vision [37]	81	22	0.59
	CRIPAC-MCT [38]	69	52	0.12
	Equalised Model [4]	NA	NA	NA
	MSMCT	75	15	0.64
Dataset 2	*hfutdspmct* [36]	2103	152	0.28
	USC-Vision [37]	95	60	0.62
	CRIPAC-MCT [38]	93	84	0.10
	Equalised Model [4]	NA	NA	NA
	MSMCT	77	53	0.67
Dataset 3	*hfutdspmct* [36]	54	84	0.03
	USC-Vision [37]	115	133	0.05
	CRIPAC-MCT [38]	61	62	0.01
	Equalised Model [4]	NA	NA	NA
	MSMCT	63	59	0.35
Dataset 4	*hfutdspmct* [36]	67	151	0.06
	USC-Vision [37]	177	115	0.34
	CRIPAC-MCT [38]	93	98	0.02
	Equalised Model [4]	NA	NA	NA
	MSMCT	149	105	0.44

TABLE V: Performance comparison using the average results of the trackers over four sub-datasets in *experiment 3*.

	Method	mme^s	mme^c	MCTA
Average	*hfutdspmct* [36]	1031	125	0.16
	USC-Vision [37]	117	82	0.40
	CRIPAC-MCT [38]	**79**	74	0.06
	Equalised Model [4]	NA	NA	NA
	MSMCT	91	**58**	**0.52**

TABLE VII: Quantitative evaluation results for the main system parts in *Experiment 3*. The constraint 1 is a system without the learned intra-camera affinity model of the tracks, while the constrains 2 refer to a system without the learned inter-camera affinity model.

Datasets	Brief Description	mme^s	mme^c	MCTA
Dataset 1	*Constraint 1*	77	18	0.61
	Constraint 2	83	24	0.58
	$OverallSystem$	75	15	0.64
Dataset 2	*Constraint 1*	99	63	0.61
	Constraint 2	102	67	0.60
	$OverallSystem$	77	53	0.67
Dataset 3	*Constraint 1*	71	69	0.23
	Constraint 2	81	74	0.19
	$OverallSystem$	63	59	0.35
Dataset 4	*Constraint 1*	143	106	0.41
	Constraint 2	154	112	0.37
	$OverallSystem$	149	105	0.44



(a) (b)

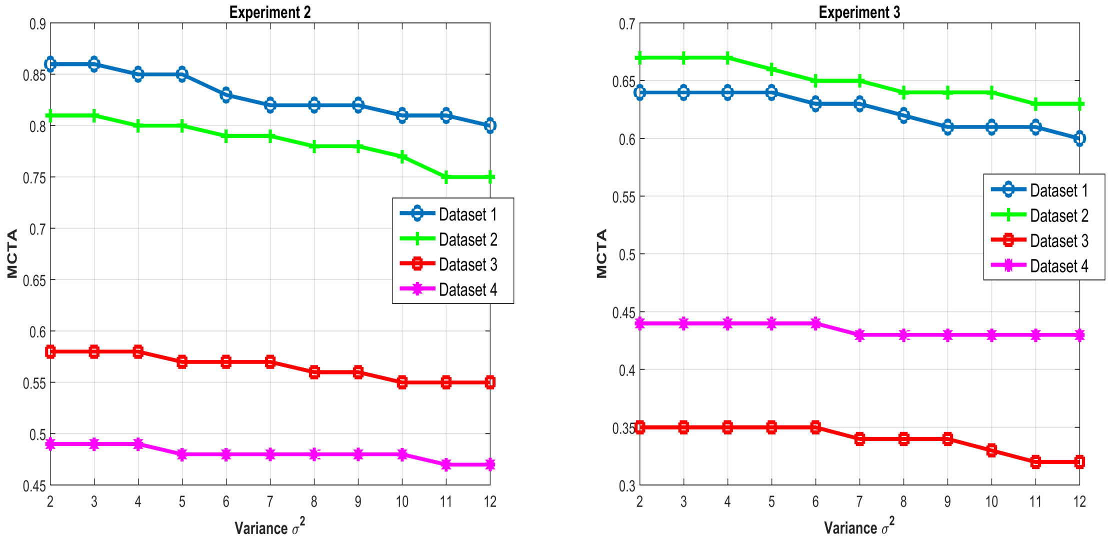
(c) (d)

Fig. 6: Sensitivity test results for the number of target mismatches in *NLPR-MCT* datasets for increasing the variance σ^2. (a, b) Sensitivity test results for the number of mismatches in a single-camera mme_t^s in *Experiment 2* and *Experiment 3*, respectively. (c, d) Sensitivity test results for the number of mismatches in cross-camera mme_t^c in *Experiment 2* and *Experiment 3*, respectively.

However, the proposed method performs on the initial tracks rather than the set of sparse target detections. Hence, it reduces the graph size and is more efficient in terms of complexity and faster from a run-time point of view. For example, the run-time of the proposed method is 20Hz and 12Hz on datasets 3 and 4 of *NLPR-MCT*, respectively. The method is implemented using MATLAB on a Xeon $3.5GHz$ CPU. Although, we used pre-computed point trajectories for the intra-camera affinity metric, the main computational complexity stems from solving the minimisation via TRW-S [35] and from learning the target-specific metrics for the targets in the inter-camera affinity metric. Speed-ups can be reached by parallel implementations of the target-specific metric learning.

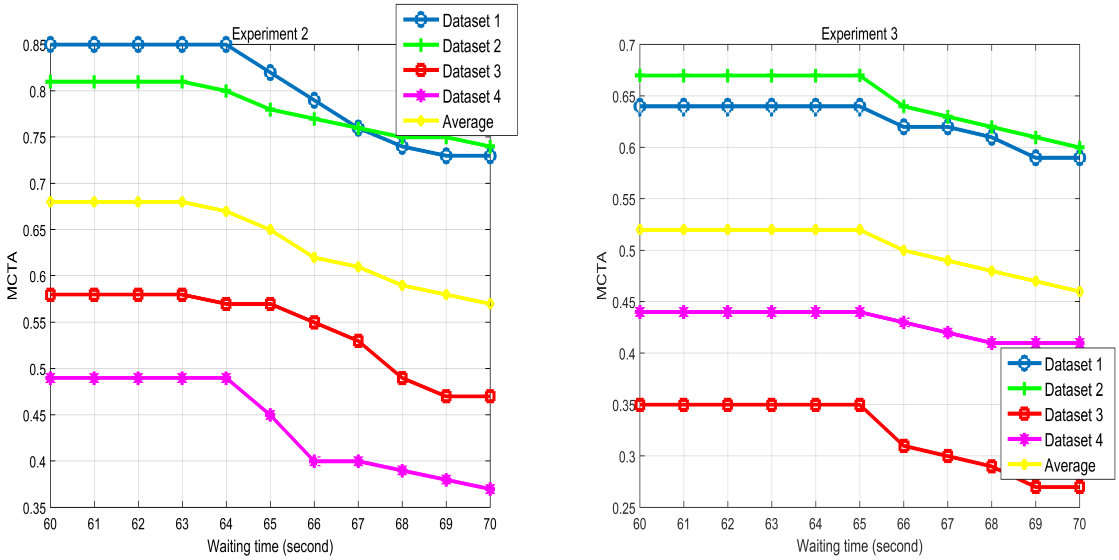
VII. Discussion

- Tables II and III show a quantitative comparison of the proposed approach to the state-of-the-art methods in two different experiments. As can be seen from Table II, except for *USC-Vision* [37], the performance of the other trackers degrades significantly. The reason is that the results of the single-camera data association, which are used as input for these trackers, are not perfect.
- Due to the benefits of the hierarchical association in *USC-Vision* [37] to build its target tracks, the rate of mismatches mme_t^s of *USC-Vision* in a single-camera is less than ours in *datasets* 1 *and* 2 (see Table II). However, under a real object detector in Table III, where there



(a) (b)

Fig. 7: Performance evaluation of the proposed tracker in *NLPR-MCT* datasets for increasing the variance σ^2. (a) Sensitivity test results in *Experiment 2*. (b) Sensitivity test results in *Experiment 3*.



(a) (b)

Fig. 8: Performance evaluation of the proposed tracker in *NLPR-MCT* datasets for increasing the waiting time $\frac{\Delta_f}{fps}$. (a) Sensitivity test results in *Experiment 2*. (b) Sensitivity test results in *Experiment 3*.

are many more false positives, we achieve a superior performance compared to other trackers due to the better formulation of data association in our proposed global optimisation. Tables IV and V demonstrate an *average* result of our tracker compared to other trackers.

- The tracking results of the proposed tracker in Tables VI and VII proves the effectiveness of the proposed affinity models. From the observations in an incomplete tracking system without using the proposed affinity model, the *MCTA* performance measure has been decreased noticeably since the discrimination power of the tracker is reduced due to the fragmented tracks and false positives.
- The value K is set to a high number, such as the maximum number of the targets in the cross-view camera scene. In fact, what happens is that the tracker will select a lot of target states (graph nodes) during the optimisation procedure with only those graph nodes set to zero by the selection matrix S that will be discarded afterwards.
- The optimisation framework is relevant to some of the state-of-the-art energy-based image co-segmentation methods [46], [47]. However, here we consider the long temporal similarities among targets to model the proposed

appearance affinity metrics.

VIII. Conclusions

We have proposed a method for multi-target tracking, where multiple target trajectories are inferred in the topology of disjoint camera views. Such a multi-person tracking method is a core ingredient for higher level group behaviour analysis methods. The proposed method commences with the initial target track and integrates both intra-camera and inter-camera data association into a single framework. At the core of the framework lies target-specific model learning for the inter-camera data association. Moreover, we consider local cues such as point trajectories to better estimate the affinity between any pairs of tracks within a single-camera view. The inference of the presented method is formulated as a global optimisation problem, which is solved through a proposed iterative approach to simultaneously find the target sates, while accounting for the occluded targets in the disjoint camera views. The conducted experiments verify that (a) the proposed affinity measure preforms remarkably better than other traditional affinity metrics and (b) a state-of-the-art performance was achieved over a wide range of benchmark instances. In particular, compared to the winner of the Multi-Camera Object Tracking Challenge [36] in 2014, the proposed approach records the highest average *MCTA* measure with a margin of $> 12\%$.

Regarding the extension of this work into a real-time application, one possible solution is using the *sliding-window* approach (e.g. of k frames), which means a sliding window of the k frames is processed temporarily to run our algorithm on these k frames, then for the next k frames and so on. Therefore, it would provide a near-online tracking algorithm with a k-frame delay. Another possible extension of the proposed tracker into a real-time scenario is to only consider the detections at the current frame (frame t) with the previous k frames ($t-k$ to $t-1$) and then perform the data association between the generated target tracks from previous k frames and the current frame detections. However, there are two main problems with this approach: first, due to too many graph nodes (including tracks in $t-k$ to $t-1$ and detections in t), the data association costs too much time and cannot meet the real-time requirement; Second, the false positive detections in real-time would cause the data association to a bad solution and an unreliable tracker. To tackle this drawback, fast and strong convolutional neural network based object detection methods, e.g. [48], can be utilised to speed up the object detection and improve the trackers accuracy attributed to the detections.

References

[1] "NLPR-MCT Dataset," http://mct.idealtest.org/Datasets.html.

[2] A. Dehghan, Y. Tian, P. H. Torr, and M. Shah, "Target identity-aware network flow for online multiple target tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1146–1154.

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

[4] W. Chen, L. Cao, X. Chen, and K. Huang, "An equalised global graphical model-based approach for multi-camera object tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, 2016, DOI: 10.1109/TCSVT.2016.2589619.

[5] S.-I. Yu, Y. Yang, and A. Hauptmann, "Harry potter's marauder's map: Localizing and tracking multiple persons-of-interest by nonnegative discretization," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013, pp. 3714–3720.

[6] W. Brendel, M. Amer, and S. Todorovic, "Multiobject tracking as maximum weight independent set," in *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2011, pp. 1273–1280.

[7] K. Shafique and M. Shah, "A noniterative greedy algorithm for multiframe point correspondence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 1, pp. 51–65, 2005.

[8] L. Zhang, Y. Li, and R. Nevatia, "Global data association for multi-object tracking using network flows," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2008, pp. 1–8.

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

[10] A. Milan, K. Schindler, and S. Roth, "Detection-and trajectory-level exclusion in multiple object tracking," in *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013, pp. 3682–3689.

[11] K. Schindler, "Continuous energy minimization for multi-target tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 1, 2014.

[12] R. Kumar, G. Charpiat, and M. Thonnat, "Multiple object tracking by efficient graph partitioning," in *Asian Conference on Computer Vision*. Springer, 2014, pp. 445–460.

[13] Y. Xiang, A. Alahi, and S. Savarese, "Learning to track: Online multi-object tracking by decision making," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4705–4713.

[14] F. Solera, S. Calderara, and R. Cucchiara, "Learning to divide and conquer for online multi-target tracking," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4373–4381.

[15] V. Kettnaker and R. Zabih, "Bayesian multi-camera surveillance," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1999.*, vol. 2. IEEE, 1999.

[16] O. Javed, Z. Rasheed, K. Shafique, and M. Shah, "Tracking across multiple cameras with disjoint views," in *Ninth IEEE International Conference on Computer Vision, 2003*. IEEE, 2003, pp. 952–957.

[17] A. R. Dick and M. J. Brooks, "A stochastic approach to tracking objects across multiple cameras," in *AI 2004: Advances in Artificial Intelligence*. Springer, 2005, pp. 160–170.

[18] D. Makris, T. Ellis, and J. Black, "Bridging the gaps between cameras," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004*, vol. 2. IEEE, 2004, pp. II–205.

[19] F. Porikli, "Inter-camera color calibration by correlation model function," in *ICIP 2003. Proceedings. 2003 International Conference on Image Processing, 2003*, vol. 2. IEEE, 2003, pp. II–133.

[20] A. Gilbert and R. Bowden, "Tracking objects across cameras by incrementally learning inter-camera colour calibration and patterns of activity," in *Computer Vision–ECCV 2006*. Springer, 2006, pp. 125–136.

[21] J. Sturges and T. Whitfield, "Locating basic colours in the munsell space," *Color Research & Application*, vol. 20, no. 6, pp. 364–376, 1995.

[22] C.-H. Kuo, C. Huang, and R. Nevatia, "Multi-target tracking by on-line learned discriminative appearance models," in *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 685–692.

[23] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, "Multicamera people tracking with a probabilistic occupancy map," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 267–282, 2008.

[24] M. Hofmann, D. Wolf, and G. Rigoll, "Hypergraphs for joint multi-view reconstruction and multi-object tracking," in *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013, pp. 3650–3657.

[25] L. Leal-Taixé, G. Pons-Moll, and B. Rosenhahn, "Branch-and-price global optimization for multi-view multi-target tracking," in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 1987–1994.

[26] W. Chen, L. Cao, J. Zhang, and K. Huang, "An adaptive combination of multiple features for robust tracking in real scene," in *2013 IEEE*

International Conference on Computer Vision Workshops (ICCVW). IEEE, 2013, pp. 129–136.

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

[28] W. Li, R. Zhao, and X. Wang, "Human reidentification with transferred metric learning," in *Asian Conference on Computer Vision*. Springer, 2012, pp. 31–44.

[29] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013, pp. 3586–3593.

[30] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2014, pp. 152–159.

[31] K. Fragkiadaki, W. Zhang, G. Zhang, and J. Shi, "Two-granularity tracking: Mediating trajectory and detection graphs for tracking under occlusions," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 552–565.

[32] B. Bozorgtabar and R. Goecke, "Efficient multi-target tracking via discovering dense subgraphs," *Computer Vision and Image Understanding*, vol. 144, pp. 205–216, 2016.

[33] T. Dietterich, R. Lathrop, and L. Perez, "Solving the Multiple Instance Problem with Axis-Parallel Rectangle," *Artificial Intelligence*, vol. 89, pp. 31–71, 1997.

[34] O. Chapelle and S. S. Keerthi, "Efficient algorithms for ranking with svms," *Information Retrieval*, vol. 13, no. 3, pp. 201–215, 2010.

[35] V. Kolmogorov, "Convergent tree-reweighted message passing for energy minimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1568–1583, 2006.

[36] "Multi-camera object tracking challenge," http://mct.idealtest.org.

[37] Y. Cai and G. Medioni, "Exploring context information for inter-camera multiple target tracking," in *2014 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2014, pp. 761–768.

[38] W. Chen, L. Cao, X. Chen, and K. Huang, "A novel solution for multi-camera object tracking," in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 2329–2333.

[39] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[40] B. Keni and S. Rainer, "Evaluating multiple object tracking performance: the clear mot metrics," *EURASIP Journal on Image and Video Processing*, vol. 2008, 2008.

[41] C.-H. Kuo and R. Nevatia, "How does person identity recognition help multi-person tracking?" in *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2011, pp. 1217–1224.

[42] A. Andriyenko and K. Schindler, "Multi-target tracking by continuous energy minimization," in *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2011, pp. 1265–1272.

[43] J. Ferryman and A. Shahrokni, "An overview of the PETS 2009 challenge," in *Eleventh IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*. IEEE, 2009.

[44] J. Domke and Y. Aloimonos, "Deformation and viewpoint invariant color histograms." in *BMVC*, 2006, pp. 509–518.

[45] M. Piccardi and E. D. Cheng, "Multi-frame moving object track matching based on an incremental major color spectrum histogram matching algorithm," in *CVPR Workshops. IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, 2005*. IEEE, 2005, pp. 19–19.

[46] Y. Zhang, X. Chen, J. Li, C. Wang, and C. Xia, "Semantic object segmentation via detection in weakly labeled video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3641–3649.

[47] W.-C. Chiu and M. Fritz, "Multi-class video co-segmentation with a generative multi-video model," in *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013, pp. 321–328.

[48] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," *arXiv preprint arXiv:1612.08242*, 2016.

Behzad Bozorgtabar is postdoctoral researcher at IBM research lab-Australia (Multimedia Analytics Group). He received his PhD in Information Science and Engineering from the University of Canberra, Australia, in 2016. He was awarded the International Postgraduate Research Scholarship for his PhD studies. His research focus are in the areas of visual tracking and video scene understanding and its applications in sports videos, and more generally in computer vision and machine learning.

Roland Goecke is Professor of Affective Computing at the University of Canberra. He is the Director of the Human-Centred Technology Research Centre and leads the Vision and Sensing Group, University of Canberra. He received his Masters degree in Computer Science from the University of Rostock, Germany, in 1998 and his PhD in Computer Science from the Australian National University, Canberra, Australia, in 2004. His research interests are in affective computing, pattern recognition, computer vision, human-computer interaction and multimodal signal processing.