PREPRINT: INFORMATION FUSION, VOL. 111, 102496, 2024, DOI: 10.1016/J.INFFUS.2024.102496

Track Initialization and Re-Identification for 3D Multi-View Multi-Object Tracking

Linh Van Ma[a], Tran Thien Dat Nguyen[b], Ba-Ngu Vo[b], Hyunsung Jang[c], Moongu Jeon[a,*]

[a] *School of Electrical Engineering and Computer Science at GIST, Gwangju, Korea*
[b] *School of Electrical Engineering, Computing and Mathematical Sciences, Curtin University, Australia*
[c] *Department of EO/IR Systems Research and Development, LIG Nex1, Korea*

Abstract

We propose a 3D multi-object tracking (MOT) solution using only 2D detections from monocular cameras, which automatically initiates/terminates tracks as well as resolves track appearance-reappearance and occlusions. Moreover, this approach does not require detector retraining when cameras are reconfigured but only the camera matrices of reconfigured cameras need to be updated. Our approach is based on a Bayesian multi-object formulation that integrates track initiation/termination, re-identification, occlusion handling, and data association into a single Bayes filtering recursion. However, the exact filter that utilizes all these functionalities is numerically intractable due to the exponentially growing number of terms in the (multi-object) filtering density, while existing approximations trade-off some of these functionalities for speed. To this end, we develop a more efficient approximation suitable for online MOT by incorporating object features and kinematics into the measurement model, which improves data association and subsequently reduces the number of terms. Specifically, we exploit the 2D detections and extracted features from multiple cameras to provide a better approximation of the multi-object filtering density to realize the track initiation/termination and re-identification functionalities. Further, incorporating a tractable geometric occlusion model based on 2D projections of 3D objects on the camera planes realizes the occlusion handling functionality of the filter. Evaluation of the proposed solution on challenging datasets demonstrates significant improvements and robustness when camera configurations change on-the-fly, compared to existing multi-view MOT solutions.

Keywords: Multi-view, Multi-sensor, Multi-object Visual Tracking, Occlusion Handling, Generalized Labeled Multi-Bernoulli, Re-Identification, Adaptive Birth.

1. Introduction

Visual tracking is a branch of multi-object tracking (MOT), which aims at estimating an unknown number of object trajectories from video sequences. There are two main approaches to MOT: track-by-detection

[*] Corresponding author

Email addresses: linh.mavan@gist.ac.kr (Linh Van Ma), t.nguyen1@curtin.edu.au (Tran Thien Dat Nguyen), ba-ngu.vo@curtin.edu.au (Ba-Ngu Vo), hyunsung.jang@lignex1.com (Hyunsung Jang), mgjeon@gist.ac.kr (Moongu Jeon)

and track-before-detect. In the former, object detection is obtained independently and then supplied to the tracker to generate track estimates, while the latter operates on the input signal without object detection. In practice, track-before-detect is computationally intensive and track-by-detection is more commonly used, especially for visual MOT due to the efficiency and reliability of 2D object detectors. The main challenges are the uncertainties in the number of objects and data association. Numerous (track-by-detection) MOT algorithms have been developed, usually under the three main paradigms: multiple hypothesis tracking (MHT) [1]; joint probabilistic data association (JPDA) [2]; and random finite set (RFS) [3].

The advancement and popularity of 2D visual MOT is mainly driven by fast and reliable 2D object detectors. When object motion is slow (relative to the frame rate) and object detection is accurate, simple trackers with kinematic/shape cues such as SORT [4] and IoU-Tracker [5] can achieve accurate tracking rate with little computation time. For challenging scenarios, with higher levels of uncertainty, more sophisticated trackers are needed [6, 7]. In addition, objects in 2D images are usually rich in visual features (e.g., pedestrians walking on the streets) and visual cues that can be exploited to distinguish different objects [8, 9], improve data association as well as re-identification of lost tracks when they re-appear [8], assuming slow variations in the visual appearance of objects.

Since objects such as people, cars, drones, etc. reside in the 3D world, 2D trajectories are not adequate for scene understanding or post-tracking analysis [10, 11], which requires 3D visual tracking. Moreover, trajectories in 3D world frame are more informative for applications such as sports analytics, age care, school environment monitoring, etc. Multi-view data also helps resolving occlusions since objects occluded in one view can be detected in other views.

A popular solution to 3D visual tracking is applying MOT to 3D detections obtained by using multi-view fusion to reconstruct objects in 3D from the 2D multi-view detections [12, 13]. However, unlike the detection of objects in 2D images, determining the 3D locations of objects from multi-view images is challenging [14, 15]. While some deep learning solutions can achieve high detection accuracy, training 3D object detectors is computationally demanding, especially for high dimensional scenarios (e.g., large number of cameras) [16]. Moreover, when the camera configurations change, the detectors need to be retrained, which limits the online operation of the tracker.

We propose a 3D visual tracking algorithm that exploits the extracted features from 2D multi-view detections via multi-sensor MOT to automatically initiates/terminates and re-identifies tracks as well resolving occlusions. Unlike many of the 3D visual tracking techniques that only provide global trajectories on the ground plane, the proposed solution processes 2D detections from multiple monocular cameras, online, to provide trajectories in 3D world frame. Our approach takes advantage of advances in 2D object detection and multi-sensor MOT that exploits geometric information from cameras with overlapping fields of view to accurately estimate the shape and position of 3D objects. The proposed multi-view MOT (MV-MOT) algorithm has a linear complexity in the number of detections across all cameras. Moreover, it does not require detector to be retrained when the cameras are reconfigured, and is amenable to seamless fusion with other types of sensor data. Performance evaluations on challenging datasets demonstrate significant improvements in tracking accuracy compared to existing solutions, and robustness when camera configurations change on-the-fly. Ablation studies are also presented to illustrate its advantages. A schematic of the proposed 3D visual tracking solution is shown in Fig. 1. Our contributions are summarized as follows:

- Novel multi-object dynamic and measurement models that jointly account for object kinematics, shapes, visual features on different cameras, and occlusion (including partial and complete occlusion);
- An approximation of the MV-MOT filter that automatically performs 3D track initialization/termination, re-identification, and occlusion handling using 2D multi-view monocular detection, with linear complexity in the number of detections across the cameras.
- Extensive experiments to evaluate the performance on challenging benchmarks including the Curtin multi-camera (CMC) [17] and WILDTRACK (WT) [13] datasets.

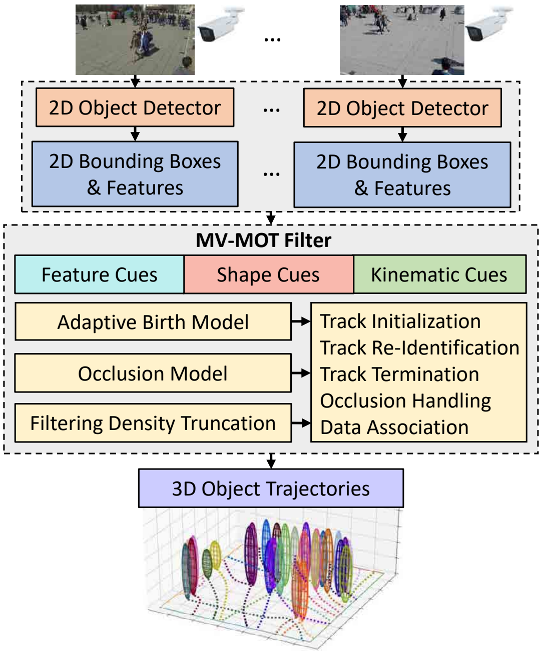

Figure 1: Schematic of the proposed 3D MV-MOT solution. Multi-view detections (bounding boxes and visual features from all cameras) is supplied to the MV-MOT filter, which integrates multi-object dynamic and measurement models to realize all MOT functionalities.

The paper is organized as follows. In Section 2 related works in 2D/3D object detection and tracking are discussed. Section 3 introduces the dynamic and measurement models together with the Bayes recursion that form our 3D visual MV-MOT solution. In Section 4, we propose an efficient approximation of the MV-MOT filter that realizes automatic track initiation/re-identification and occlusion resolution. Extensive experiments to verify the effectiveness of our tracking solutions are given in Section 5, and Section 6 concludes the paper.

2. Related Works

2.1. Visual Multi-Object Detection

Multi-object detection from 2D images is a key research topic in computer vision. Early detectors use template matching to localize objects in images [18]. Many learning-based solutions rely on trainable classifiers such as support vector machines or Adaboost to detect objects [19] using features such as Haar, scale-invariant feature transform (SIFT) [20], and histogram of oriented gradients (HOG) [21]. Deep learning has become popular in object detection due to the utility of convolutional neural networks (CNNs) [22]. Combinations of effective region-proposal algorithms [23] and CNN features (i.e., features extracted from CNNs) have resulted in real-time, high-performance 2D object detectors [24]. YOLO algorithms that bypass the region-proposal step by casting detection as a regression problem are significantly more efficient [25]. Recently, algorithms that formulate object detection as a set of learning tasks have also been proposed [26]. Publicly available large-scale datasets have been instrumental for the fast-paced development of 2D object detection solutions, especially learning-based methods [27, 28].

Detecting occluded objects in 2D images is a challenging problem. Multi-view images provide more accurate detection than single-view images by fusing information from different views. In [29], a probabilistic occupancy map is constructed from background-subtracted images to locate objects on the ground plane. However, this method tends to generate a high number of false alarms. Nonetheless, this could be reduced by the Bayesian network-based technique proposed in [30]. Alternatively, Gibbs sampling is used in [31] to generate the number of objects and their spatial locations from a posterior (conditioned on 2D detections). CNN features can also be used for multi-view detection, e.g., in [16] a discriminative CNN feature extraction module is used in conjunction with a generative occlusion model to construct an existence probability map of objects on the ground plane, while in [32], CNN features are projected to the ground plane and then fed into classifiers to localize objects. Methods based on similar projections are also proposed in [33, 34].

2.2. Visual Multi-Object Tracking

Visual MOT solutions can be categorized as online or batch. Batch algorithms estimate object trajectories from a batch of data, with computational complexities per time step growing over the time window. On the other hand, online algorithms estimate object trajectories at each time step when new data arrive, with computational complexities per time step that are independent of time, and hence are preferred over batch algorithms in practice. In 2D visual MOT, algorithms that exploit only motion and shape information are fast but cannot handle complex scenarios [5, 4]. Alternatively, visual features of objects can be used to improve tracking. Hand-crafted features (e.g., SIFT, HOG) are not effective in distinguishing different objects [35], and CNN features are more suitable due to the multi-scale representation. In [35] separate models for detection and feature extraction are used. While using a single model for both tasks was demonstrated to have better efficiency in [36], a balance between the two tasks needs to be considered [9]. State-of-the-art (SOTA) 2D multi-object trackers that utilize feature cues in the literature include POI [37], MOTDT [35], DeepSORT [38], and GSDT [39].

Multi-view MOT solutions are becoming increasingly important due to the proliferation of cameras and the better tracking performance over single-view techniques. Homography constraints are used to track human feet in [40], while in [41] heads are localized in single-view images and then transformed to world coordinates to perform tracking. In [42], principal axes are used to associate tracklets between cameras, while in [43] advanced semantic cues are used. In [44], 2D detections are mapped into 3D positions, and then combined with relevant cues (i.e., motion, features, geometry proximity) to associate tracklets using a hierarchical composition model. In [45, 46], 3D objects of different classes are tracked with a 3D track query model using multiple monocular cameras for autonomous driving applications.

Occlusion handling is an important functionality of visual tracking. In the single-view case, certain solutions rely on detectors that can localize parts of the objects [47], albeit training such detectors to yield accurate localization results is difficult. A popular approach is to use designated modules that analyze occlusion, using object depth [48], or spatial information of objects and their interactions to resolve occlusion [49, 50]. In the multi-view case, occlusion can also be implicitly resolved in the multi-view data fusion process, usually exploiting object locations, either at the detection or tracking step [16, 17].

Data association is a crucial and challenging problem in track-by-detection MOT. Simple algorithms such as the global nearest neighbor (GNN) [2] consider a single hypothesis of data association. More sophisticated MOT frameworks such as MHT, JPDA, and RFS have demonstrated improved tracking performance by keeping multiple data association hypotheses. The labeled RFS solutions [51], such as the GLMB filter, are well suited for online and multi-view MOT due to the low-complexity and efficiency [52]. Indeed, the GLMB filter has been used in various computer vision problems [6, 7], including multi-sensor data association for multi-view occlusion handling [17].

While MOT functionalities such as data association, track initiation/termination, re-identification, and occlusion handling are captured in the GLMB filtering recursion [52], an exact implementation realizing all these functionalities is numerically intractable. In [17], an approximation was developed to address occlusion, but the re-identification functionality was neglected, and track initiation requires a combination of accurate prior birth models (which is not always available) with clustering. While object features improve tracking performance [8, 9], they have not been exploited by the filter to improve data association and resolve track re-identification. Moreover, the occlusion model in [17] does not account for partial occlusions, and thus was unable to exploit SOTA 2D object detection technique [53]. Without accurate prior information, initializing tracks from multi-sensor measurements is challenging due to the unknown number of new tracks, miss-detection, false alarms, and the large number of possible detection combinations from multiple sensors. The recent solution in [54] utilizes Monte Carlo (MC) technique to initialize tracks where existence probabilities depend on their measurement likelihood and how likely the detections are already associated with known tracks. While, this solution can be directly applied to visual tracking, it is difficult to find a balance between speed and accuracy not to mention the inability resolve track appearance-reappearance.

3. Bayesian Multi-View MOT

This section presents a Bayesian tracker that can handle all functionalities of a multi-view multi-object tracker from automatic track initialization, termination, re-identification to multi-view data association and occlusion handling. In particular, details on the object dynamic and measurement models will be given together with a Bayes recursion that propagates the multi-object density over time. Notations commonly used in this paper are tabulated in Tab. 1.

Table 1: List of symbols.

Notation	Description
$\otimes$	Kronecker product (for matrices)
h^X	$\prod_{x\in X} h(x)$ with $h^{\emptyset} = 1$
$\langle f, g\rangle$	$\int f(x)g(x)dx$, inner product of f and g,
$j : k$	$j, j+1, ..., k$
$x^{(j:k)}$	$x^{(j)}, x^{(j+1)}, ..., x^{(k)}$
$x_{j:k}$	$x_j, x_{j+1}, ..., x_k$
$\mathbb{X}$	single object state space
$\mathbb{L}$	discrete label space
$\mathbb{B}$	discrete label space for new birth objects
$\boldsymbol{X}$	labled multi-object state
$\mathcal{L}(\boldsymbol{X})$	set of labels of multi-object state $\boldsymbol{X}$
$\boldsymbol{x} = (x, \ell)$	labeled single-object state (with label ℓ)
$\boldsymbol{\pi}$	multi-object density
Ω	MS/MV-GLMB recursion operator
$\{(r_B^{(\ell)}, p_B^{(\ell)})\}_{\ell\in\mathbb{B}}$	parameters of new birth objects
$f_{S,+}(x_+\|x, \ell)$	single-object transition density
$P_{S,+}(\boldsymbol{x})$	survival probability of labeled state $\boldsymbol{x}$
$\mathbb{Z}^{(c)}$	measurement space of camera c
$Z^{(c)}$	set of measurements of camera c
$z^{(c)}$	single-view measurement of camera c
$g^{(c)}(z^{(c)}\|\boldsymbol{x})$	single-object single-view measurement likelihood function for camera c
$g^{(c)}(Z^{(c)}\|\boldsymbol{X})$	multi-object single-view measurement likelihood function for camera c
$\boldsymbol{g}\,(Z\|\boldsymbol{X})$	multi-object multi-view measurement likelihood function
$P_D^{(c)}(\boldsymbol{x}, \boldsymbol{X})$	detection probability for camera c
$\alpha^{(\ell,c)}$	observed feature of object ℓ at camera c
$\Phi^{(c)}(x)$	box bounding an object with state x in camera c's image plane
$\gamma^{(c)}$	association map for camera c
γ	multi-view association map
$\Gamma^{(c)}$	space of association maps for camera c
Γ	space of multi-view association maps
$\mathcal{L}_{\gamma^{(c)}}$	live label set of association map $\gamma^{(c)}$
$\delta_Y[X]$	generalized Kronecker delta function, takes on 1 if $X = Y$, and 0 otherwise
$\mathcal{N}(.; \mu, P)$	Gaussian pdf with mean μ and covariance P

3.1. Object Dynamic Model

The state $\boldsymbol{x} = (x, \ell)$ of an object consists of attribute x from an attribute space $\mathbb{X}$ and a label ℓ from a discrete label space $\mathbb{L}$. An object born at time k, is assigned a time-invariant label $\ell = (k, \iota)$, where ι is a unique index to differentiate objects born at the same time. The attribute x consists of 3D position ζ, 3D velocity $\dot{\zeta}$, and shape parameter ς. The *multi-object state* at a given time k is a *finite set of individual object states* in $\mathbb{X} \times \mathbb{L}$ with distinct labels [51].

At time k, a set (possibly empty) of new objects is born. The set of all possible labels of object born at time k is a subset of $\mathbb{L}$, denoted by $\mathbb{B}$. A new object with label ℓ is born with probability $r_B^{(\ell)}$, and conditional on which its attribute is distributed according to $p_B^{(\ell)}$. The birth parameters $\{(r_B^{(\ell)}, p_B^{(\ell)})\}_{\ell \in \mathbb{B}}$ could be provided apriori (if statistics of newborns are known), or estimated from the data.

Given a multi-object state $\boldsymbol{X}$ at time k, each $(x, \ell) \in \boldsymbol{X}$ either survives to the next time with probability $P_{S,+}(x, \ell)$ or dies with probability $1 - P_{S,+}(x, \ell)$. Conditional on survival the object takes on the new state (x_+, ℓ_+) according to the transition density $f_{S,+}(x_+|x, \ell)\delta_\ell[\ell_+]$ [51], where the generalized Kronecker delta $\delta_\ell[\ell_+]$, defined to be 1 when $\ell = \ell_+$ and 0 otherwise, ensures the label remains unchanged. The multi-object state $\boldsymbol{X}_+$ at the next time is the superposition of newborns and surviving objects, and is distributed according to the *multi-object Markov transition density* $\boldsymbol{f}_+(\boldsymbol{X}_+|\boldsymbol{X})$ (an explicit expression is not needed in this work, nonetheless it can be found in [51]). Hereon, we use the subscript '+' to indicate the next time.

In this work, we use the survival probability model proposed in [6]. The shape parameter ς is a triplet of (logarithms of) the half-lengths of the principal axes of the ellipsoid containing the object, and follows a random-walk model. The kinematics $(\zeta,\ \dot{\zeta})$ follows a nearly constant velocity model. Specifically, given the current attribute x, the next attribute x_+ is distributed by [17]

$$f_{S,+}(x_+|x, \ell) = \mathcal{N}\left(x_+; Fx + b, Q\right), \tag{1}$$

where
$$F = \begin{bmatrix} I_3(T) & 0_{6\times 3} \\ 0_{3\times 6} & I_3 \end{bmatrix}, I_3(T) = I_3 \otimes \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, b = \begin{bmatrix} 0_{6\times 1} \\ -\upsilon^{(\varsigma)}/2 \end{bmatrix},$$
$$Q = \begin{bmatrix} V(\upsilon^{(\zeta)}, T) & 0_{6\times 3} \\ 0_{3\times 6} & \mathrm{diag}(\upsilon^{(\varsigma)}) \end{bmatrix}, V(\upsilon^{(\zeta)}, T) = \mathrm{diag}(\upsilon^{(\zeta)}) \otimes \begin{bmatrix} \frac{T^2}{2} \\ T \end{bmatrix} \begin{bmatrix} \frac{T^2}{2} & T \end{bmatrix},$$

T is the sampling period, $\upsilon^{(\zeta)}$ and $\upsilon^{(\varsigma)}$ are 3D vectors of noise variances for the position and shape parameter, respectively. The Gaussianity of the logarithms of the half-lengths ensures that they are non-negative. This is equivalent to log-normal distributions of these half-lengths with unit-mean and variances $e^{\upsilon_i^{(\varsigma)}}$-1, i=$1, 2, 3$ [17].

3.2. Multi-View Measurement Model

Given cameras $1, ..., C$ and a multi-object state $\boldsymbol{X}$, each $\boldsymbol{x} \in \boldsymbol{X}$ is detected by camera c with probability $P_D^{(c)}(\boldsymbol{x}; \boldsymbol{X})$ and generates the single-view measurement $z^{(c)} \in \mathbb{Z}^{(c)}$ ($\mathbb{Z}^{(c)}$ is the measurement space of the camera c) with likelihood $g^{(c)}(z^{(c)}|\boldsymbol{x})$, or miss-detected with probability $1 - P_D^{(c)}(\boldsymbol{x}; \boldsymbol{X})$. While the detection probability is assumed independent of the other (or all) objects in most MOT algorithms, this assumption is not valid in occlusions. The objects in $\boldsymbol{X} \backslash \{\boldsymbol{x}\}$ could occlude $\boldsymbol{x}$, which translates to a low detection probability for $\boldsymbol{x}$. Thus a suitable detection probability model that accounts for occlusion is needed for occlusion handling [17].

3.2.1. Single-View Single-Object Measurement Model

Conditional on detection by camera c, $\boldsymbol{x}$ is observed as a 2D bounding box and a feature vector, i.e., $z^{(c)} = (z_p^{(c)}, z_e^{(c)}, z_f^{(c)})$ where $z_p^{(c)}$ is the box center, $z_e^{(c)}$ is its extent (parameterized by the logarithms of the width and height in the camera c's image plane), and $z_f^{(c)}$ is the feature vector (pertaining to appearance or identity). Since the kinematic and feature observations of an object are independent, the single-view single-object measurement likelihood $g^{(c)}(z^{(c)}|\boldsymbol{x})$ can be written as

$$g^{(c)}(z_p^{(c)}, z_e^{(c)}, z_f^{(c)}|\boldsymbol{x}) = g_b^{(c)}(z_p^{(c)}, z_e^{(c)}|x, \ell) g_f^{(c)}(z_f^{(c)}|\ell), \tag{2}$$

where $g_b^{(c)}$ and $g_f^{(c)}$ are, respectively, the bounding box and feature measurement likelihoods.

The bounding box measurement $(z_p^{(c)}, z_e^{(c)})$ is a noisy version of the box $\Phi^{(c)}(x)$ bounding the image of object (x, ℓ) in camera c's image plane, which can be computed analytically via the projection matrix, see [55]. Hence, likelihood of $(z_p^{(c)}, z_e^{(c)})$ is given by [17]

$$g_b^{(c)}(z_p^{(c)}, z_e^{(c)}|x, \ell) = \mathcal{N}\left(\begin{bmatrix} z_p^{(c)} \\ z_e^{(c)} \end{bmatrix}; \Phi^{(c)}(x), \operatorname{diag}\left(\begin{bmatrix} \upsilon_p^{(c)} \\ \upsilon_e^{(c)} \end{bmatrix}\right)\right), \tag{3}$$

where $\upsilon_p^{(c)}$ and $\upsilon_e^{(c)}$ are the noise variances for the center and the extent (in logarithm) of the box, respectively.

The feature measurement vector $z_f^{(c)}$ captures the object's visual appearance, e.g., color histograms, HSV features, Deep Learning features. Visual features can be used to identify objects since they are relatively stable [9] or slowly varying with time [36]. Nonetheless, visual features can suddenly change [4], and are not always reliable [9]. Thus, visual features models usually accommodate several modes of observation [4]. Without loss of generality, we use a likelihood for $z_f^{(c)}$ with two modes, a strong mode to capture the stable slow variation, and a weaker mode to capture the sudden changes. Specifically,

$$g_f^{(c)}(z_f^{(c)}|\ell) \propto \sigma s_f(z_f^{(c)}, \alpha^{(\ell,c)}) + \bar{\sigma} s_f(z_f^{(c)}, \bar{\alpha}^{(\ell,c)}), \tag{4}$$

where: s_f is a non-negative function that monotonically increases with the similarity between its arguments; $\alpha^{(\ell,c)}$ and $\bar{\alpha}^{(\ell,c)}$ are, respectively, the nominal feature vectors for the stable and unstable modes, with respective weights σ and $\bar{\sigma}$. Further, following [36], the slow variation of the feature vector is modeled by adaptively updating the nominal feature at each time step via

$$\alpha_{+}^{(\ell,c)} = \vartheta_0 \alpha^{(\ell,c)} + (1-\vartheta_0) z_f^{(c)},$$

where ϑ_0 is a weight that controls the contribution of the observed data to the nominal feature. In essence, $\alpha^{(\ell,c)}$ is the exponential moving average of the observed feature with momentum ϑ_0. The initial the feature $\alpha^{(\ell,c)}$, of object ℓ at camera c, can take on the feature computed from the measurement that it is initialized with or some prior value if it is initially misdetected.

3.2.2. Single-View Multi-Object Measurement Model

The measurement set $Z^{(c)}$ from camera c is a superposition of object-originated measurements and independent false positives (or clutter). Conditional on the multi-object state $\boldsymbol{X}$, the object-originated measurements are statistically independent [51]. False positives are commonly parameterized by an intensity function $\kappa^{(c)}$, where the number of false positives is Poisson distributed with mean $\langle \kappa^{(c)}, 1 \rangle$, and individual false positives are independent and identically distributed according to $\kappa^{(c)}/\langle \kappa^{(c)}, 1 \rangle$, where $\langle f, g \rangle = \int f(x) g(x) dx$. In most MOT algorithms $\kappa^{(c)}$ is often assumed constant and known apriori. Nonetheless, it can also be estimated on-the-fly along with the multi-object state, albeit with additional computations [56].

To account for unknown data association, it is necessary to consider different object-to-measurement mappings. At time k, an *association map* for camera c is a mapping $\gamma^{(c)} : \mathbb{L} \to \{-1 : |Z^{(c)}|\}$ such that each label can only be mapped to at most one measurement, where $|Z^{(c)}|$ denotes the cardinality of $Z^{(c)}$ [51]. For a label ℓ, $\gamma^{(c)}(\ell) = -1$ represents a non-existent object, $\gamma^{(c)}(\ell) = 0$ represents a miss-detection at camera c, while $\gamma^{(c)}(\ell) > 0$ represents the scenario that ℓ generates measurement $z^{(c)}_{\gamma^{(c)}(\ell)}$ at camera c. Let $\Gamma^{(c)}$ denote the set of all association maps, $\mathcal{L}(\boldsymbol{X})$ the set of labels of multi-object state $\boldsymbol{X}$, and $\mathcal{L}_{\gamma^{(c)}} \triangleq \left\{\ell : \gamma^{(c)}(\ell) \geqslant 0\right\}$ is the live label set of $\gamma^{(c)}$. Then, the *single-view multi-object measurement likelihood* for camera c is given by [51]

$$\boldsymbol{g}^{(c)}(Z^{(c)}|\boldsymbol{X}) \propto \sum_{\gamma^{(c)} \in \Gamma^{(c)}} \delta_{\mathcal{L}(\gamma^{(c)})}[\mathcal{L}(\boldsymbol{X})] \Big[\psi^{(c,\gamma^{(c)}(\mathcal{L}(\cdot)))}_{Z^{(c)},\boldsymbol{X}}(\cdot)\Big]^{\boldsymbol{X}}, \tag{5}$$

where $\delta_A[B] = 1$ if $A = B$ and zero otherwise,

$$\psi^{(c,j)}_{\{z^{(c)}_{1:|Z^{(c)}|}\},\boldsymbol{X}}(\boldsymbol{x}) = \begin{cases} 1 - P_D^{(c)}(\boldsymbol{x};\boldsymbol{X}), & j = 0 \\ \frac{P_D^{(c)}(\boldsymbol{x};\boldsymbol{X}) g^{(c)}(z_j^{(c)}|\boldsymbol{x})}{\kappa^{(c)}(z_j^{(c)})}, & j > 0 \end{cases}. \tag{6}$$

3.2.3. Multi-View Multi-Object Measurement Model

Noting that $\gamma^{(1)}(\ell) = ... = \gamma^{(C)}(\ell) = -1$ if ℓ does not exist, we define a *multi-view association map* as a tuple $\gamma \triangleq (\gamma^{(1:C)})$ of association maps such that, $\gamma^{(c)}(\ell) = -1$ for any c implies, $\gamma^{(c)}(\ell) = -1$ for all c. This means $\gamma : \mathbb{L} \to \{-1\}^C \uplus (\mathbb{J}^{(1)} \times \cdots \times \mathbb{J}^{(C)})$, where $\mathbb{J}^{(c)} \triangleq \{0 : |Z^{(c)}|\}$. Let Γ denote the space of multi-view association maps, $Z \triangleq (Z^{(1:C)})$, and assuming that conditional on $\boldsymbol{X}$, these constituent sets are mutually independent, then the *multi-view multi-object measurement likelihood* is given by [52]:

$$\boldsymbol{g}\left(Z|\boldsymbol{X}\right) \propto \sum_{\gamma\in\Gamma} \delta_{\mathcal{L}_\gamma}[\mathcal{L}\left(\boldsymbol{X}\right)] \left[\psi_{Z,\boldsymbol{X}}^{(\gamma(\mathcal{L}(\cdot)))}\left(\cdot\right)\right]^{\boldsymbol{X}}, \tag{7}$$

where $\mathcal{L}_\gamma \triangleq \{\ell : \gamma^{(1)}(\ell), ..., \gamma^{(C)}(\ell) \geqslant 0\}$ denotes the *live label set* of the multi-view association map γ, and

$$\psi_{Z,\boldsymbol{X}}^{(j^{(1:C)})}\left(\boldsymbol{x}\right) \triangleq \prod_{c=1}^{C} \psi_{Z^{(c)},\boldsymbol{X}}^{(c,j^{(c)})}\left(\boldsymbol{x}\right). \tag{8}$$

3.3. Bayesian Multi-View MOT Filter

In Bayesian estimation, the *multi-object filtering density* is the probability density of the current multi-object state conditioned on the observation history. It encapsulates all statistical information on the multi-object state, given the observed data, and prior information described by the multi-object transition density $\boldsymbol{f}(\cdot|\cdot)$ and observation likelihood $\boldsymbol{g}(\cdot|\cdot)$. Multi-object state/trajectory estimate can be determined from the multi-object filtering density via the Joint Multi-object (JoM) or Marginal Multi-object (MaM), including labeled-MaM, estimators [57], [58]. The latter are commonly used due to their computational tractability. The MaM/labeled-MaM estimate is the most probable (or expected) multi-object state given the most probable cardinality/label-set [57], [58].

The multi-object filtering density $\boldsymbol{\pi}$ can be propagated forward to the next time via the Bayes recursion

$$\boldsymbol{\pi}_{+}(\boldsymbol{X}_{+}) \propto \boldsymbol{g}(Z_{+}|\boldsymbol{X}_{+}) \int \boldsymbol{f}_{+}(\boldsymbol{X}_{+}|\boldsymbol{X})\boldsymbol{\pi}(\boldsymbol{X})\delta\boldsymbol{X}. \tag{9}$$

This approach is not only applicable to objects with independent motion, and detection observations, but for more general models including cell mitosis [7], social force model [59], track-before-detect [6], as well as merged measurements [60]. It also offers the capability to fuse different measurement types, e.g., track-before-detect measurement with detections, simply by multiplying their likelihoods.

The (exact) Bayes MOT filter (9) would fulfill all MOT functionalities. The integration of suitable multi-object dynamic and observation models into the multi-object filtering density allows the filter to initiate/terminate/re-identify tracks, resolve multi-view data association and occlusions, from the observed data. Unfortunately, exact implementation is intractable due mainly to the exponential growth in memory requirement and computational resources. Existing approximations, designed for speed in generic applications, impede MOT functionalities such as track initiation/re-identification and occlusion resolution.

4. Approximate MV-MOT Filter

This section presents an approximate Multi-View MOT (MV-MOT) filter that realizes automatic track initiation/re-identification and occlusion resolution by using an adaptive birth model that accounts for reappearing objects and a high-fidelity geometric occlusion model. In Subsection 4.1, we present a commonly used approximation to the Bayes MV-MOT filter (9), which involves a generalized labeled multi-Bernoulli (GLMB) approximation for analytical tractability, and truncating the resulting GLMB components for numerical tractability [17]. A high-fidelity yet tractable occlusion model based on projections of 3D objects on the camera planes that accommodates full/partial occlusions is developed in Subsection 4.2. In Subsection 4.3, we detail an adaptive birth model to realize track initiation and rectify the GLMB truncation to realize re-identification.

4.1. Multi-View GLMB Recursion

This subsection outlines the two-step approximation of the Bayes MV-MOT filter. Consider first the approximation of the multi-object filtering density $\boldsymbol{\pi}$, by a GLMB of the form

$$\hat{\boldsymbol{\pi}}\left(\boldsymbol{X}\right) = \delta_{|\boldsymbol{X}|}\left[\left|\mathcal{L}\left(\boldsymbol{X}\right)\right|\right]\sum_{I,\xi}\omega^{(I,\xi)}\delta_{I}\left[\mathcal{L}\left(\boldsymbol{X}\right)\right]\left[p^{(\xi)}\right]^{\boldsymbol{X}}, \tag{10}$$

where: $I \in \mathcal{F}(\mathbb{L})$, the *class of all finite subsets of* $\mathbb{L}$; $\xi \in \Xi$, the *space of multi-view association map histories* $\gamma_{1:k}$; each $\omega^{(I,\xi)}$ is a non-negative weight such that $\Sigma_{I,\xi}\omega^{(I,\xi)} = 1$; and each $p^{(\xi)}(\cdot,\ell)$ is a probability density on $\mathbb{X}$. The weight $\omega^{(I,\xi)}$ can be interpreted as the probability of *hypothesis* (I,ξ), and conditional on (I,ξ), $p^{(\xi)}(\cdot,\ell)$ is the probability density of the attribute of $\ell \in I$. A GLMB is completely characterized by its parameters, and hence we adopt the abbreviation

$$\hat{\boldsymbol{\pi}} = \left\{(\omega^{(I,\xi)}, p^{(\xi)}) : (I,\xi) \in \mathcal{F}(\mathbb{L}) \times \Xi\right\}. \tag{11}$$

Remark 1: Note that the GLMB cardinality distribution, from which we determine the most probable cardinality n^* for the MaM estimator, is given by

$$\mathrm{Prob}(|\boldsymbol{X}| = n) = \sum_{I,\xi}\omega^{(I,\xi)}\delta_n[|I|].$$

For efficiency, instead of computing the most probable multi-object state, we compute the estimated states from $p^{(\xi^*)}(\cdot,\ell)$ for each $\ell \in I^*$, where (I^*,ξ^*) is the most probable hypothesis such that $|I^*| = n^*$.

The class of GLMBs is a versatile family of multi-object densities due to its closure under the Bayes recursion (9) and efficient approximation (linear complexity in the total number of detections across the sensors) for commonly used multi-object system models [52]. Specifically, for $P_D^{(1:C)}\left(\boldsymbol{x};\boldsymbol{X}\right) = P_D^{(1:C)}\left(\boldsymbol{x}\right)$, if the multi-object filtering density at the current time is a GLMB, then it is also a GLMB at the next time, and given by the MS-GLMB recursion [52]

$$\widehat{\boldsymbol{\pi}}_{+} \quad = \quad \Omega_{+}(\widehat{\boldsymbol{\pi}}; P_{D,+}^{(1:C)}, \boldsymbol{f}_{B,+}),$$

where $\boldsymbol{f}_{B,+} \triangleq \{(r_{B,+}^{(\ell)}, p_{B,+}^{(\ell)})\}_{\ell \in \mathbb{B}_{+}}$ denotes the parameters of the birth model[1]. While the number of GLMB components grows exponentially with time, they can be truncated with minimum L_1-error, using multi-dimensional rank assignment [61] or Gibbs sampling [62]. Unfortunately, when $P_D^{(1:C)}(\boldsymbol{x}; \boldsymbol{X}) \neq P_D^{(1:C)}(\boldsymbol{x})$ as per our occlusion model, $\boldsymbol{\pi}_{+}$ is not a GLMB, and is computationally intractable in general.

An approximate multi-view GLMB (MV-GLMB) filter for occlusion models with a general $P_D^{(1:C)}(\boldsymbol{x}; \boldsymbol{X})$ has been developed in [17] by combining piecewise approximation of $P_D^{(1:C)}(\boldsymbol{x}; \boldsymbol{X})$ with importance sampling via the Gibbs sampler. The approximate GLMB filtering density is propagated using the MV-GLMB recursion

$$\widehat{\boldsymbol{\pi}}_{+} \quad = \quad \widehat{\Omega}(\widehat{\boldsymbol{\pi}}; P_{D,+}^{(1:C)}, \boldsymbol{f}_{B,+}), \tag{12}$$

summarized in Alg. 2 of [17], which extends the MS-GLMB filter to address $P_D^{(1:C)}(\boldsymbol{x}; \boldsymbol{X}) \neq P_D^{(1:C)}(\boldsymbol{x})$.

Remark 2: The GLMB filtering density can be further approximated by retaining only the best component after each filtering cycle. This approximation uses only the most likely (multi-sensor) measurement-to-track assignment, which is conceptually similar to the strategy of the global nearest-neighbour (GNN) tracker [2]. Although this results in considerable improvement in processing speed, performance is expected to degrade, especially in low signal-to-noise scenarios (see also the ablation study in Subsection 5.3.2).

Fig. 2 illustrates how our new adaptive birth model and occlusion model is integrated into the MV-GLMB filter to realize the MOT functionalities of (automatic) Track Inititialization, Track Re-Identification, Track Termination, and Occlusion Handling. Details on the proposed occlusion model will be given in the next subsection, and the adaptive estimation of the birth model parameters $\{(r_B^{(\ell)}, p_B^{(\ell)})\}_{\ell \in \mathbb{B}_{+}}$ will be given in Subsection 4.3.

4.2. Occlusion Modeling

Rather than using an external occlusion handling module to provide better tracks, the Bayes MOT filter accounts for occlusion via an occlusion model, described by the detection probability of the objects. In the presence of occlusions, the more accurate model is, the better the tracking results. An occlusion model was proposed in [17], where the detection probabilities of objects in the shadow regions of others (w.r.t. the

[1] The MS-GLMB recursion also depends on the measurements and other multi-object system parameters, but we suppress them for clarity.

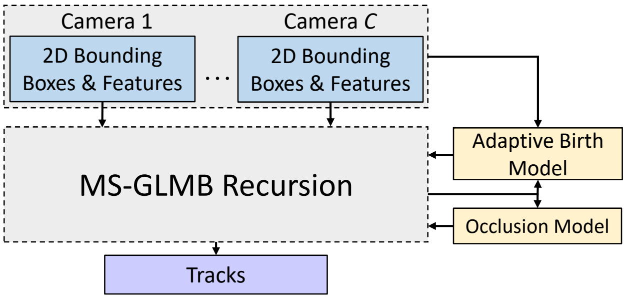

Figure 2: Schematic of the proposed multi-view MOT filter, with Adaptive Birth Model and Occlusion Model that realize MOT functionalities.

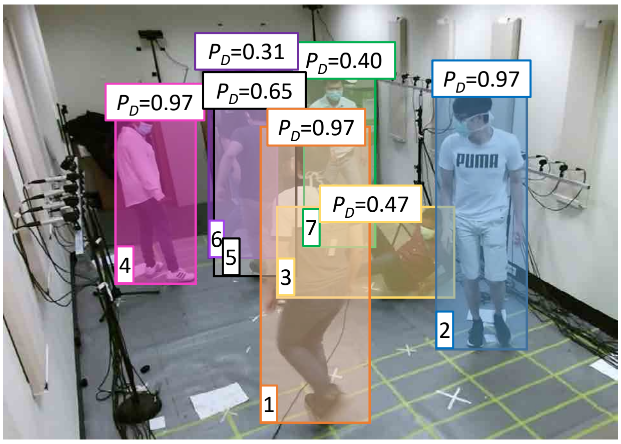

Figure 3: For illustration, tracks are indexed from the closest to the furthest from the camera. Track 4 has no overlap with any other tracks and thus has maximum detection probability. Tracks 1 and 2 overlap with other tracks, but closer to the camera (i.e., lower bottom corner), hence they also have maximum detection probability. Track 6 overlaps with track 5, but track 5 has higher detection probability, because it is closer to the camera.

LoS of the camera) are assigned small values. While this model is accurate for full occlusions, it does not address partial occlusions, where the detector still has a high probability of detecting the objects.

In this subsection, we present a new occlusion model that accommodates partial (and full) occlusions. Our model is based on the proportion of area overlap between the boxes bounding the images of the occluded and occluding objects on camera's image plane. The larger the overlap, the lower the detection probability of the occluded object. Noting that an object can only be occluded by those in front of it (i.e., closer to the camera), let $\mathrm{Fr}^{(c)}(\boldsymbol{x};\boldsymbol{X})$ denote the subset of objects in $\boldsymbol{X}$ that are in front of $\boldsymbol{x} \in \boldsymbol{X}$ with respect to camera c. Then the occlusion score of $\boldsymbol{x}$ is given by

$$O^{(c)}\left(\boldsymbol{x};\boldsymbol{X}\right) = \frac{\mathrm{Area}\left(\Phi^{(c)}(\boldsymbol{x}) \bigcap \left(\bigcup_{\boldsymbol{x}' \in \mathrm{Fr}^{(c)}(\boldsymbol{x};\boldsymbol{X})} \Phi^{(c)}(\boldsymbol{x}')\right)\right)}{\mathrm{Area}\left(\Phi^{(c)}(\boldsymbol{x})\right)}, \tag{13}$$

where $\Phi^{(c)}(\boldsymbol{x}) = \Phi^{(c)}(x)$ for $\boldsymbol{x} = (x, \ell)$, and $\mathrm{Area}(S)$ is the area of a 2D shape S. Since the more occluded the object is the less likely it will be detected, we use the following the detection probability

$$P_D^{(c)}\left(\boldsymbol{x};\boldsymbol{X}\right)=\max(\epsilon,1-\epsilon-O^{(c)}\left(\boldsymbol{x};\boldsymbol{X}\right)), \tag{14}$$

so as to cap it between ϵ and $1-\epsilon$. Some example detection probability values for a given camera according to this model is shown in Fig. 3.

Remark 3: The subset $\mathrm{Fr}^{(c)}(\boldsymbol{x};\boldsymbol{X})$ can be determined by comparing the distances of the objects from camera c. Alternatively, assuming all objects are on the same ground level, we can compare the lower bottom corners of the bounding boxes of the objects on camera c's plane: those with lower bottom corners are closer to the camera.

4.3. Adaptive Birth Modeling

While the MV-GLMB filter can provide automatic track initiation and re-identification, it requires a combination of accurate prior birth model (that varies with time) and prudent approximation. In this subsection, we develop a tractable technique to estimate birth model online and rectify the GLMB truncation process to realize track initiation and re-identification.

4.3.1. Adaptive Birth Model Parameters

In [54], an efficient technique was developed for estimating the LMB birth model parameters $\{(r_{B,+}^{(\ell_+)}, p_{B,+}^{(\ell_+)})\}_{\ell\in\mathbb{B}_+}$ (Section 3.1), using the current multi-sensor measurement. This approach seeks an empirical LMB birth model that provides a good fit of the multi-camera measurement Z. Given the current GLMB filtering density (11), suppose that the multi-camera measurement Z is generated from new birth objects according to the multi-view association map $\mathring{\gamma}:\mathbb{B}_+\rightarrow\{-1\}^C\uplus(\mathbb{J}^{(1)}\times\cdots\times\mathbb{J}^{(C)})$. Then, the best fitting empirical LMB model is given by [54]

$$\{(\widehat{r}_{\mathring{\gamma}}^{(\ell_+)},\widehat{p}_{\mathring{\gamma}}^{(\ell_+)})\}_{\ell_+\in\mathcal{L}_{\mathring{\gamma}}}, \tag{15}$$

where: $\mathcal{L}_{\mathring{\gamma}}$ is the live label set of $\mathring{\gamma}$;

$$\begin{aligned}
\widehat{r}_{\mathring{\gamma}}^{(\ell_+)} &= \min\left(r_{B,+}^{*},\frac{\lambda_{B,+}r_U\left(\mathring{\gamma}(\ell_+)\right)\bar{\psi}_{Z,B}^{(\mathring{\gamma}(\ell_+))}(\ell_+)}{\left\langle r_U\left(\mathring{\gamma}(\cdot)\right),\bar{\psi}_{Z,B}^{(\mathring{\gamma}(\cdot))}(\cdot)\right\rangle}\right);\\
\widehat{p}_{\mathring{\gamma}}^{(\ell_+)}(x_+) &\propto \int f_+(x_+|x,\ell_+)p_{B,0}^{(\ell_+)}(x)\psi_{Z,B}^{(\mathring{\gamma}(\ell_+))}(x,\ell_+)dx;
\end{aligned}$$

$r_{B,+}^{*}$ is a prescribed maximum birth probability; $\lambda_{B,+}$ is a prescribed expected number of births;

$$r_U(j^{(1:C)}) = \prod_{c=1}^{C}\left[1-\sum_{I,\xi}1_{\xi^{(c)}(I)}(j^{(c)})\omega^{(I,\xi)}\right]; \tag{16}$$

$$\bar{\psi}_{Z,B}^{(j^{(1:C)})}(\ell) = \langle p_{B,0}(\cdot,\ell),\psi_{Z,B}^{(j^{(1:C)})}(\cdot,\ell)\rangle; \tag{17}$$

$\xi^{(c)}(I) = \{\gamma^{(c)}(\ell) : \ell \in I\}$; $p_{B,0}(x,\ell)$ is a prescribed prior birth probability density; $\psi_{Z,B}^{(j^{(1:C)})}(x,\ell)$ is $\psi_{Z,\boldsymbol{X}}^{(j^{(1:C)})}(x,\ell)$ in (7) with $P_D^{(c)}(\boldsymbol{x};\boldsymbol{X})$ set to a prescribed (constant) detection probability $P_{D,B}^{(c)}$, and the feature likelihood $g_f^{(c)}(z_f^{(c)}|\ell)$ in the single-view single-object measurement likelihood (2) set to a uniform distribution (hence, only the bounding box measurements are used in the birth model estimation).

The empirical LMB birth (15) is completely parameterized by the multi-view association map $\mathring{\gamma}$. This birth model reduces prior knowledge on a large number of LMB model parameters to only four prescribed parameters $r^*_{B,+}$, $\lambda_{B,+}$ (usually set to 1), $p_{B,0}(x,\ell)$ and $P_{D,B}$. Intuitively, the componenst of a good fitting empirical LMB should have significant existence probabilities.

Remark 4: Note that in this work, we used the multi-view association map $\mathring{\gamma} = (\mathring{\gamma}^{(1:C)})$ instead of the injection $\theta_B : \mathbb{J}^{(1)} \times \cdots \times \mathbb{J}^{(C)} \to \mathbb{B}_+$ in [54]. The key difference is that $\mathring{\gamma}$ constrains each single-camera detection to originate from at most one object (see Subsection 3.2.2). In contrast, θ_B relaxes this constraint and allows each single-camera detection to originate from multiple objects, which can result in increased false track initiations, especially for scenarios with large areas. Nonetheless, this relaxation enables the authors to develop a Gibbs sampler to compute a good fitting (empirical LMB parameterized by) θ_B [54].

Since the Gibbs sampler in [54] cannot accommodate the constraint of at most one object per detection, we use clustering to determine a good fitting (empirical LMB parameterized by) $\mathring{\gamma}$. Intuitively, the detections generated by the same object at every camera would be clustered around the object's position when projected into the ground plane. Hence, the $\mathring{\gamma}$ constructed by clustering (single-camera) detections in the ground plane so that each cluster corresponds to detections generated by an object, provides a good fit to the multi-camera measurement Z. Note that since $\mathring{\gamma}$ is multi-camera association map, each detection can only belong to at most one cluster.

The clustering algorithm is described in Alg. 1. The multi-view association map $\mathring{\gamma}$, represented as an assignment matrix where each row consists of the measurement indices that belong to one cluster. In step one, a set of initial cluster means is generated in a similar manner to the popular mean shift clustering algorithm. In step two, $\mathring{\gamma}$ is constructed by sequentially appending each row of the associated measurement indices. In the pseudocode, the 'TransformToGroundPlane' function is a homography transformation taking 2D measurements to their ground plane positions. The 'dist' function computes the distance between points in the ground plane. The 'ComputeCentroid' function returns the centroid in the ground plane with inputs as a list of points and the corresponding indices specifying which points are used to compute the centroid.

4.3.2. Track Initialization and Re-Identification

The birth model enables the MV-GLMB recursion (12) to automatically initiates new tracks, and in principle, re-identify reappearing tracks. Labels that have ever existed (up to the current time) are captured in some components of the (untruncated) GLMB filtering density. When new data arrives, the MV-GLMB recursion updates their existence probabilities accordingly so that those in the scene have high existence probabilities and vice-versa. However, in practice, component truncation deletes labels with prolonged low existence probabilities permanently from the GLMB density. This means they cannot be recovered even when new data support their reappearance, and each LMB birth parameter in $\{(\widehat{r}^{(\ell)}_{\mathring{\gamma}}, \widehat{p}^{(\ell)}_{\mathring{\gamma}})\}_{\ell\in\mathcal{L}_{\mathring{\gamma}}}$ could either correspond to a new track, or reappearing track.

To restore the filter's track re-identification functionality, we propose to retain tracks that would have been deleted in the the GLMB truncation, herein referred to as *Tentatively Terminated* (TT) tracks, and relabel the subset of $\{(\widehat{r}^{(\ell)}_{\mathring{\gamma}}, \widehat{p}^{(\ell)}_{\mathring{\gamma}})\}_{\ell\in\mathcal{L}_{\mathring{\gamma}}}$ with the labels of the TT tracks that best match them in visual features[2]. A TT track retains the visual feature from its corresponding label in highest weighted GLMB component at the time of TT, and will only be permanently deleted if it is not re-identified within a prescribed period. Note that after relabeling we remove the corresponding TT track from the TT set. The GLMB recursion with the relabeled LMB birth model will update the existence probabilities of the reappearing and new birth tracks in accordance with the received multi-camera data.

Optimal assignment can be used to match the live labels $\mathcal{L}_{\mathring{\gamma}}$ and the TT tracks, and only those with matching scores above a recall threshold τ_R are relabeled. The matching score of a label $\ell_+ \in \mathcal{L}_{\mathring{\gamma}}$ to a TT label $\ell = (s, \iota)$ with feature $\alpha^{(\ell,c)}$ from each camera $c \in \{1 : C\}$, is defined as

$$R_{\ell_+,\ell} = \frac{k-s}{e(\ell)} \max_{c\in\{1:C\}} s_f\left(f(z^{(c)}_{\mathring{\gamma}^{(c)}(\ell_+)}), \alpha^{(\ell,c)}\right), \tag{18}$$

where $e(\ell)$ denotes the number of times that label ℓ is included in the GLMB density but not as a TT track, $f(z^{(c)}_j)$ denotes the feature component of the j-th measurement from camera c, and $s_f(\cdot,\cdot)$ is the similarity measure between two feature vectors, see also (12). The rationale behind the time ratio in (18) is that the longer the label exists in the GLMB density, the more likely it still exists even though it is TT.

5. Experimental Results

This section presents experimental evaluations of our proposed 3D MV-MOT algorithm, referred to as Multiview GLMB-Adaptive Birth (MV-GLMB-AB) filter, on the WILDTRACK (WT) [13] dataset (with 7 cameras) and the Curtin multi-camera (CMC) dataset (with 4 cameras, and five sequences CMC1-5) [17]. While the WT dataset is popular for 3D MOT, it only provides the *true ground plane* positions of the objects, *not their 3D positions* and extents. The CMC dataset fills this gap, with details on height, width, and 3D position. Sequences 1-5 of the CMC dataset have different object densities for performance evaluation with varying levels of difficulties. Further, sequences 4 and 5 also include jumping and falling people to evaluate pose estimation capability.

[2] Visual features are better suited for re-identification because they are relatively stable over time [9], whereas kinematic and shapes attributes vary with time, while tracks reappear almost independently from the locations where they disappeared.

Algorithm 1: Clustering for Adaptive Birth.

```
**Input**: $Z$, $h_{1:C}$, $\epsilon$
**Output**: $\mathring{\gamma}$

*Step 1: Generate Initial Cluster Means*
$S = \varnothing$
**for** $c = 1 : C$ **do**
    $S^{(c)} = \text{TransformToGroundPlane}(Z^{(c)})$
    $S = S \cup S^{(c)}$
**while** *all elements in* $S$ ***not*** *converge* **do**
    **for** $g = 1 : |S|$ **do**
        **if** $S[g]$ ***not*** *converge* **then**
            $P = \varnothing$;
            **for** $c = 1 : C$ **do**
                **for** $i = 1 : |S^{(c)}|$ **do**
                    **if** $dist(S^{(c)}[i], S[g]) < h_c$ **then**
                        $P = P \cup S^{(c)}[i]$
            $new\_S_g \leftarrow K\,(\min(P) - S[g])\min(P)$
            **if** $dist((S[g], new\_S_g) < \epsilon$ **then**
                $S[g]$ is converge
            $S[g] \leftarrow new\_S_g$
*Step 2: Generate Multi-View Association Map*
$\mathring{\gamma} = \varnothing$; $t = 0$;
**for** $c = 1 : C$ **do**
    **for** $i = 1 : |S^{(c)}|$ **do**
        $l = -1$; $p = +\infty$; $t = t + 1$;
        **for** $j = 1 : NumberOfRow(\mathring{\gamma})$ **do**
            $cen = \text{ComputeCentroid}(\mathring{\gamma}[j, :],\ S^{(1:C)})$
            $d = \text{dist}\left(S_i^{(c)}, cen\right)$;
            **if** $(d < h_c) \wedge (d < p) \wedge (\mathring{\gamma}[j, c] = 0)$ **then**
                $l = j$; $p = d$;
        **if** $l = -1$ **then**
            $M = 0_{1\times C}$; $M_c = i$;
            $\mathring{\gamma} = \text{AppendRow}(\mathring{\gamma}, M)$;
        **else**
            $\mathring{\gamma}[l, i] = i$;
```

To quantify tracking performance, we use the track identity measure [63], including IDF1 score, numbers of mostly tracked (MT) tracks, partially tracked (PT) tracks, and ID switches (IDS). We also report the CLEAR MOT measure [64], including multi-object tracking accuracy (MOTA) score, numbers of false positive (FP), false negative (FN), and OSPA$^{(2)}$ error [65, 66]. For evaluations that only considered ground plane positions of the tracks, the Euclidean distance is used as the base-distance. For evaluations of 3D trajectories with extents (a 3D ellipsoid), the GIoU distance (normalized to the interval [0,1]) between 3D bounding boxes is used as the base-distance. The evaluation threshold is 1 meter for the Euclidean distance and 0.5 for the GIoU distance. The cut-off distance for the OSPA$^{(2)}$ metric is set to 1 meter for Euclidean base-distance and to 1 for GIoU base-distance. Since the filters involve random sampling, we evaluate their performance over 25 Monte Carlo (MC) runs and report the mean and standard deviation of the results.

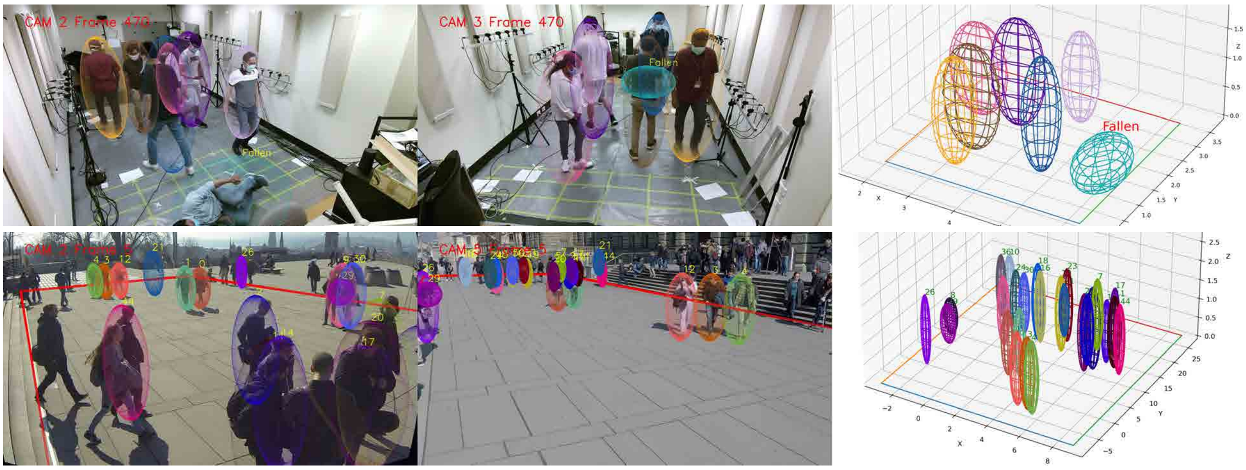

Figure 4: 3D ellipsoid estimates from the proposed MV-GLMB-AB filter using CSTrack detection inputs. Top row: CMC5 dataset at frame 470 for cameras 2 and 3. Bottom row: WT dataset at frame 25 for cameras 2 and 5 (only objects inside the red boundary are considered). The first two columns show the projected 3D estimates on the respective camera planes, and the last column shows the 3D estimates. Each color corresponds to a unique object ID. Videos are also provided in the supplementary materials.

In subsection 5.1 a comparison of tracking performance and run-time between the MV-GLMB-AB filter and other state-of-the-art methods, demonstrating the ability to re-identify tracks and uninterrupted operations when cameras are added, removed, or repositioned on-the-fly. Subsection 5.2 benchmarks the tracking performance of MV-GLMB-AB filter against baseline single-sensor filters that process ground plane measurements from ideal detectors, trained on the ground truth dataset. An ablation study on the models used in the MV-GLMB-AB filter is presented in Subsection 5.3.

The following model parameters are used for all experiments in this section. The dynamic noise variance (measured in squared meters) is set to $\upsilon^{(\zeta)}$=$[0.0012, 0.0012, 0.0012]^T$ for the CMC dataset and $\upsilon^{(\zeta)}$=$[0.0225, 0.0225, 0.0225]^T$ for the WT dataset, while $\upsilon^{(\varsigma)}$=$[0.0036, 0.0036, 0.0004]^T$ for both. For each camera, the measurement noise variance (measured in pixels) is set to $\upsilon_p^{(c)} = [400, 400]^T$, and $\upsilon_e^{(c)}$=$[0.00995, 0.0025]^T$ for upright objects and $\upsilon_e^{(c)}$=$[0.0025, 0.00995]^T$ for fallen objects. Since we observe a low number of false positive measurements in the tested scenarios, we set the clutter rate to 5 for all sensors in our implementation. If a higher number of false positive measurements is observed, the clutter rate can be estimated on-the-fly from the data.

5.1. Performance Evaluation

5.1.1. Fixed multi-camera configuration

We compare the proposed filter with current 3D MOT algorithms that process 2D multi-view detections, namely the MV-GLMB [17] and MS-GLMB [52] filters. Note that the CMC4-5 sequences are used to compare the performance on scenarios involving human poses (upright or fallen) with the MV-GLMB filter.

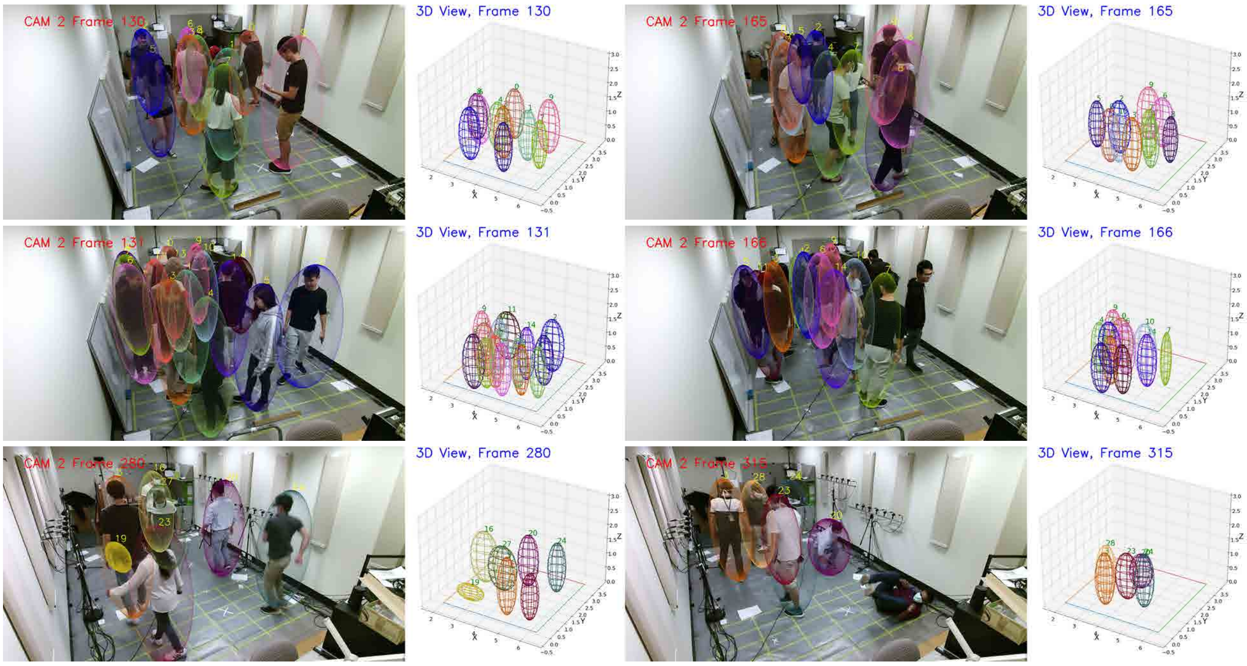

Figure 5: Track re-identification: 3D ellipsoid estimates from the MV-GLMB-AB filter using CSTrack detection. Object disappearance-reappearance (in CMC) is simulated by turning off all cameras mid-scene for 30 frames. Top row: CMC2-all cameras off from frames 130-160. Middle row: CMC3-all cameras off from frames 131-161. Bottom row: CMC5-all cameras off from frames 280-310. Columns 1 and 2 show estimates, projected on camera 2 and in 3D, before turning off all cameras. Columns 3 and 4 show the estimates 5 frames after all cameras are turned back on.

The proposed filter's output on typical scenarios in the CMC5 sequence and the WT dataset with CSTrack detections are shown in Fig. 4. For the CMC5 sequence, the proposed MOT filter yields accurate object positions and poses. The relatively poor detection quality of the CSTrack detector (see Tab. 6) on the WT dataset is manifested in several misdetections in the proposed filter's output.

Fig. 5 illustrates the filter's re-identification capability, where object disappearance-reappearance is simulated by turning off all cameras mid-scene for 30 frames so that most or all of the tracks are terminated before the cameras are on again. Lost 3D tracks are re-identified after they reappear using feature information from monocular camera images. Recall performance is best when the features are stable and unique across frames as per CMC1 and CMC2 (see also video supplementary material). Recall is poor when the objects have similar appearance (i.e., non-unique features) as per CMC3, or change their appearance quickly (i.e., unstable features) as per CMC5 with pose changes.

Quantitative comparison with other multi-view MOT algorithms are shown in Tab. 2 for the WT dataset, Tab. 3 for the CMC dataset, and Tab. 4 for the CMC dataset with all cameras turned off mid-scene to assess re-identification. The higher MOTA, IDF1 scores, and lower OSPA$^{(2)}$ errors, indicate superior performance of the proposed MOT filter (note that there is a very small performance difference between the two different implementations of the adaptive birth model, which will be discussed in the ablation study). For the CMC

Table 2: Tracking performance (in the *ground plane*) on the WT dataset with the CSTrack detector: MC means and 1 standard deviation (shown in parenthesis, only reported for the main measures). The best result for each sequence is **Bolded**.

Detector	Tracker	MT↑	PT↓	ML↓	FP↓	FN↓	IDs↓	MOTA↑	IDF1↑	OSPA$^{(2)}$↓
CSTrack	Ours	**101**	**134**	**77**	5040	**2804**	**273**	**14.7(3.37)**	**50.9(2.11)**	**0.79(0.01)**
	MV-GLMB	57	152	103	3916	4161	422	10.7(2.13)	33.2(1.07)	0.86(0.01)
	MS-GLMB	58	153	100	**3890**	4148	419	11.1(2.06)	33.3(0.97)	0.86(0.01)
FairMOT	Ours	**119**	**127**	**66**	3399	**2463**	**215**	**36.1(2.60)**	**58.4(1.98)**	**0.73(0.01)**
	MV-GLMB	45	147	120	**3237**	4353	383	16.2(1.79)	31.8(0.77)	0.86(0.00)
	MS-GLMB	42	146	123	3260	4404	379	15.5(1.48)	31.4(0.8)	0.87(0.00)

Table 3: Tracking performance (in *3D and ground plane*) on the CMC dataset with the CSTrack detector: MC means and 1 standard deviation (shown in parenthesis, only reported for the main measures). The best result for each sequence is **Bolded**.

Seq.	Tracker	Evaluation with 3D ellipsoid estimates						Evaluation with ground plane estimates					
		FP↓	FN↓	IDs↓	MOTA↑	IDF1↑	OSPA$^{(2)}$↓	FP↓	FN↓	IDs↓	MOTA↑	IDF1↑	OSPA$^{(2)}$↓
CMC1	Ours	**0**	4	**0**	**99.4(0.00)**	**99.7(0.00)**	**0.3(0.00)**	**0**	4	**0**	**99.4(0.00)**	**99.7(0.00)**	**0.07(0.00)**
	MV-GLMB	47	3	**0**	92.1(2.86)	96.0(1.49)	0.83(0.02)	47	**2**	1	92.2(2.89)	96.0(1.50)	0.78(0.02)
	MS-GLMB	20	**2**	**0**	96.4(1.97)	98.1(1.11)	0.82(0.01)	20	**2**	**0**	96.5(1.98)	98.1(1.11)	0.76(0.02)
CMC2	Ours	**18**	**36**	**5**	**97.1(1.43)**	**90.2(6.62)**	**0.41(0.03)**	**16**	33	**4**	**97.4(1.38)**	**94.0(4.28)**	**0.26(0.05)**
	MV-GLMB	353	41	56	78.3(2.76)	52.4(6.79)	0.88(0.02)	337	**26**	46	80.2(2.86)	64.4(5.92)	0.87(0.02)
	MS-GLMB	141	126	78	83.3(2.71)	47.8(5.06)	0.88(0.02)	105	89	57	87.8(1.99)	60.9(4.71)	0.87(0.02)
CMC3	Ours	**57**	**99**	**15**	**93.9(1.15)**	**78.6(5.03)**	**0.45(0.03)**	**28**	70	**14**	**96.0(1.02)**	**85.8(3.63)**	**0.33(0.04)**
	MV-GLMB	572	137	105	71.2(3.61)	43.8(3.76)	0.86(0.02)	489	**55**	87	77.6(2.88)	58.0(3.18)	0.85(0.02)
	MS-GLMB	315	327	142	72.2(4.94)	38.7(2.59)	0.89(0.01)	181	192	110	82.9(4.63)	55.6(2.65)	0.89(0.01)
CMC4	Ours	**0**	**9**	**0**	**97.5(0.29)**	**98.7(0.16)**	**0.24(0.00)**	**0**	**9**	**0**	**97.8(0.00)**	**98.9(0.00)**	**0.11(0.00)**
	MV-GLMB	19	94	3	70.9(4.00)	66.6(3.31)	0.71(0.05)	8	83	4	76.0(3.30)	68.8(3.16)	0.67(0.06)
	MS-GLMB	70	103	3	56.2(15.93)	74.6(6.90)	0.66(0.06)	60	92	4	61.1(15.79)	76.9(6.77)	0.62(0.07)
CMC5	Ours	**83**	**328**	**27**	**88.1(0.69)**	**50.9(3.02)**	**0.87(0.02)**	**32**	**277**	**24**	**91.0(0.51)**	**51.9(3.04)**	**0.86(0.02)**
	MV-GLMB	601	597	102	65.0(7.06)	24.7(2.47)	0.93(0.01)	411	408	95	75.4(6.73)	31.7(2.60)	0.94(0.01)
	MS-GLMB	456	690	155	65.0(6.53)	20.3(3.32)	0.96(0.01)	210	444	144	78.5(5.47)	27.7(4.14)	0.97(0.01)

dataset, we excluded the MT, PT, and ML scores since they give no useful insight given the small number of objects in the scenes. The poorer performance of the MV-GLMB and MS-GLMB filters arises from poor track initiation/re-identification and occlusion handling (both filters cannot re-identify tracks, and the MS-GLMB filter does not account for occlusions). This can be seen from the OSPA$^{(2)}$ error curves in Fig. 6 (the error at time k is computed over the window from the initial time to time k). In CMC1, the error increases when the cameras are turned off and decreases when the cameras are turned on, indicating correct re-identifion. In CMC2 and CMC4, although most tracks are correctly re-identified, the error does not decrease after the cameras are turned on because some tracks are assigned incorrect IDs. In CMC3 and CMC5, due to the high object density and severe occlusion, the features are unstable. Consequently, only a small number of tracks are recalled, and hence the OSPA$^{(2)}$ error increase.

Tab. 5 shows the average run-time in FPS (frame per second, on a desktop with an Intel(R) Core(TM) i7-7700K CPU @ 4.20GHz Processor without any GPU accelerations), for each 3D MV-MOT methods in the WT and CMC datasets. The proposed filter shows improved processing speed compared to MV-GLMB [17], and are able to track objects on-line. Although there are only 3 objects in CMC4, the processing time increases since we also estimate object poses. The run-time on the WT dataset is higher than that on the

Table 4: Tracking performance (in *3D and ground plane*) on the CMC dataset with disappearing-reappearing objects, and CSTrack detector: MC means and 1 standard deviation (shown in parenthesis, only reported for the main measures). Object disappearance-reappearance is simulated by turning off all cameras mid-scene for 30 frames. The best result for each sequence is **Bolded**

Seq.	Tracker	Evaluation with 3D ellipsoid estimates						Evaluation with ground plane estimates					
		FP↓	FN↓	IDs↓	MOTA↑	IDF1↑	OSPA$^{(2)}$↓	FP↓	FN↓	IDs↓	MOTA↑	IDF1↑	OSPA$^{(2)}$↓
CMC1	Ours	**0**	**85**	**0**	**87.0(0.14)**	**93.1(0.10)**	**0.38(0.00)**	**0**	**85**	**0**	**87.0(0.17)**	**93.1(0.10)**	**0.19(0.00)**
	MV-GLMB	44	92	3	78.6(2.56)	51.2(0.93)	0.93(0.01)	41	89	4	79.5(2.60)	51.2(0.91)	0.91(0.01)
	MS-GLMB	25	91	4	81.5(2.08)	51.9(0.69)	0.93(0.00)	23	89	4	82.0(2.00)	51.9(0.68)	0.91(0.01)
CMC2	Ours	**48**	**290**	**18**	**82.8(0.58)**	**74.0(6.01)**	**0.54(0.04)**	**13**	**254**	**16**	**86.3(0.7)**	**80.1(4.37)**	**0.45(0.05)**
	MV-GLMB	334	360	75	63.0(3.12)	36.5(2.43)	0.94(0.01)	279	306	64	68.7(2.79)	43.3(2.09)	0.93(0.01)
	MS-GLMB	170	445	105	65.3(2.28)	32.8(1.90)	0.95(0.00)	102	377	86	72.7(1.13)	41.0(1.87)	0.95(0.00)
CMC3	Ours	**110**	**446**	**39**	**78.9(1.05)**	**55.4(2.83)**	**0.69(0.02)**	**19**	**355**	**37**	**85.4(0.71)**	**63.0(2.48)**	**0.64(0.02)**
	MV-GLMB	556	524	132	57.1(3.20)	37.9(2.88)	0.93(0.01)	428	396	107	67.0(2.77)	44.4(2.37)	0.91(0.01)
	MS-GLMB	293	696	153	59.6(3.62)	34.9(2.18)	0.94(0.01)	162	566	126	69.7(2.96)	42.6(2.46)	0.93(0.01)
CMC4	Ours	**1**	**92**	**1**	**76.7(0.51)**	**73.9(3.91)**	**0.63(0.04)**	**0**	**91**	**1**	**77.2(0.19)**	**74.2(3.82)**	**0.56(0.05)**
	MV-GLMB	17	201	6	44.2(1.80)	55.7(0.79)	0.86(0.01)	9	193	7	48.1(1.06)	56.7(0.67)	0.84(0.02)
	MS-GLMB	37	213	5	36.5(12.69)	50.4(7.10)	0.87(0.02)	29	205	6	40.4(11.17)	52.3(6.52)	0.85(0.02)
CMC5	Ours	**89**	**506**	**34**	**83.0(0.39)**	**39.1(2.41)**	**0.92(0.01)**	**29**	**446**	**33**	**86.3(0.34)**	**40.5(2.31)**	**0.91(0.01)**
	MV-GLMB	526	822	106	60.8(7.72)	21.7(3.78)	0.96(0.01)	331	627	99	71.5(7.58)	26.7(3.26)	0.96(0.01)
	MS-GLMB	482	955	158	57.0(5.84)	18.1(1.67)	0.97(0.00)	227	701	152	70.9(5.43)	23.7(2.30)	0.98(0.00)

CMC dataset due to the higher number of objects.

Table 5: True number of objects in the sequences and MC means (1 standard deviation is shown in parenthesis) of run-time, in FPS, for different filters. The '∗' indicate our filter, and the best result for each row is **Bolded**.

Seq.	No. Obj.	MV-GLMB	MV-GLMB-AB*
WT	24	0.02 (0.18)	**0.06(0.01)**
CMC1	3	0.62 (1.53)	**28.5(0.12)**
CMC2	10	0.04 (0.41)	**7.0(0.33)**
CMC3	15	0.05 (1.51)	**4.6(0.11)**
CMC4	3	0.07 (0.09)	**3.6(0.42)**
CMC5	7	0.02 (0.06)	**2.7(0.07)**

5.1.2. Multi-Camera Reconfiguration

Similar to the MV-GLMB filter, our proposed filter only requires once-off training of the monocular detectors, allowing seamless operation without any interruption when cameras are added, removed, or repositioned. To demonstrate this capability, we construct, for each CMC sequence, a scenario involving five configurations over time, see Fig. 7. The OSPA$^{(2)}$ tracking error is benchmarked against the ideal case (baseline) where all cameras are on for the entire period in Fig. 7.

In CMC1, the different camera configurations exhibit similar performance to the baseline due to the low object density (relative to the number of cameras). In CMC2-3, the performance degrades slightly when fewer cameras are on since it becomes harder to resolve occlusion with higher object density. Although the object density is low in CMC4, tracking upright and fallen people is more challenging due to the increased uncertainty. Hence, there is a slight performance degradation relative to the baseline. CMC5 also involves tracking upright and fallen people, but at a higher object density than CMC4. As a result the baseline

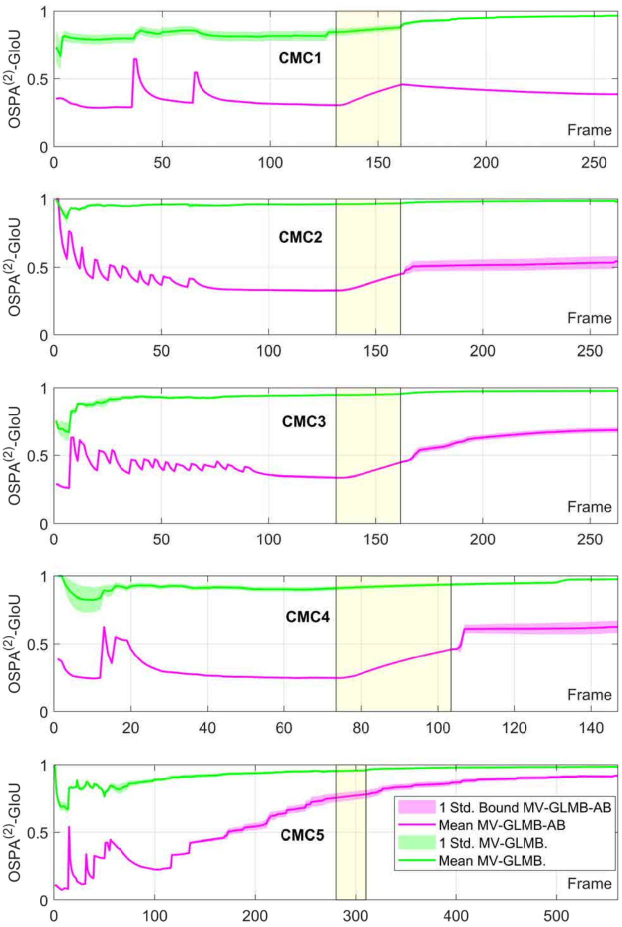

Figure 6: Track re-identification: OSPA$^{(2)}$ error of the proposed filter with CSTrack detection. Object disappearance-reappearance in the CMC dataset is simulated by turning off all cameras mid-scene for 30 frames (indicated in yellow). Tracking errors for MV-GLMB almost saturate at the maximum value. Except in CMC5 where re-identification fails because the features are not stable, the proposed filter has considerably lower tracking errors at all times.

and the reconfigured scenario tracking errors are similarly high due to the high object density (relative to the number of cameras). These results demonstrate that the proposed methods can adapt to camera reconfigurations on-the-fly without sacrificing the tracking performance. More details on tracking results can be found in the videos in the supplementary materials.

5.2. Benchmarking Against Ideal Trackers

In this study, we benchmark our 3D MV-MOT filter against the best possible track-by-detection performance via the combination of an ideal 3D detector with some of the best known (single-sensor) MOT algorithms. For the ideal detector we use the 3DROM detector [67], trained on 90% of the WT dataset[3], which is almost perfect since it is trained directly on the ground truth, and is far better than the CSTrack/FairMOT

[3] Note that the WT data only provide ground truth in the ground plane, and while it is called 3DROM, this detector only provides detections in ground plane, not in full 3D.

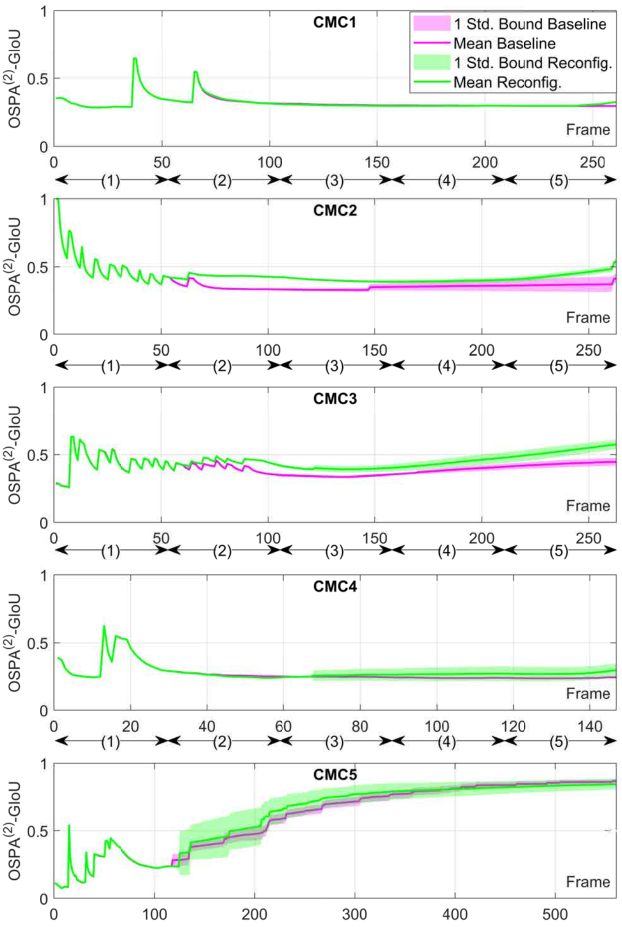

Figure 7: Multi-camera reconfiguration: OSPA$^{(2)}$ error of the proposed filter with CSTrack detection. Configuration (1): all cameras at positions 1, 2, 3, and 4 are on. Configuration (2): three cameras on at positions 2, 3, and 4. Configuration (3): three cameras on at random positions. Configuration (4): two cameras on at positions 1 and 3. Configuration (5): two cameras on at positions 2 and 4.

detector used for MV-MOT, see Tab. 6. The baseline single-sensor MOT algorithms include the GLMB [51], MHT, JPDA, GNN filters [2], and the KSP-ptracker [13] based on the DeepOcclusion detector [16]. However, since the detection is built into the tracker, we cannot evaluate the detection quality of the DeepOcclusion detector independently. The results in Tab. 7 show that the gaps in tracking performance are not as wide as the gaps in detection performance. Keeping in mind that the 3D detection input for the single-sensor filters are effectively ground truths, it is surprising that the proposed MV-MOT filter shows comparable performance to some of the ideal filters in certain measures.

Table 6: Detection quality for the WT dataset, ‘∗’ indicates 3D.

Detector	MODA↑	MODP↑	Rcll↑	Prcn↑
3DROM*	93.50	75.90	96.20	97.20
FairMOT	28.92	65.46	69.61	63.11
CSTrack	11.67	64.56	70.27	54.53

Table 7: Tracking performance (in the *ground plane*) of our filter with CSTrack detections and single-sensor filters with ideal 3D detections, on the WT dataset. The best result for each column is **Bolded**.

Detector	Tracker	MT↑	PT↓	ML↓	FP↓	FN↓	IDs↓	MOTA↑	IDF1↑	OSPA$^{(2)}$↓
CSTrack	MV-GLMB-AB*	101	134	77	5040	2804	273	14.7(3.37)	50.9(2.11)	0.79(0.01)
FairMOT	MV-GLMB-AB*	119	127	66	3399	2463	215	36.1(2.60)	58.4(1.98)	0.73(0.01)
3DROM	GLMB	**167**	107	39	**136**	**1501**	181	**81.6**	**86.4**	**0.19**
	MHT	41	125	147	502	4083	266	51.0	50.2	0.31
	JPDA	171	85	57	1522	1770	368	63.0	60.1	0.39
	GNN	168	82	63	1911	1801	489	57.6	55.4	0.52
DeepOcclusion	KSP-ptracker	72	**74**	**25**	2007	5830	**103**	72.2	78.4	0.75

5.3. Ablation Study

5.3.1. Sensitivity to Occlusion Model

In this study, we assess the effect of using object features with various occlusion models, and compare the two implementations of the adaptive birth model. In particular, we compare the tracking performance of our occlusion model (IoA), the line-of-sight model (LoS) [17], and the constant detection probability model, with and without object features on the CMC5 sequence. Tab. 8 indicates that the best performance (in IDF1 score and OSPA$^{(2)}$ error) is the combined use of object feature and the IoA occlusion model, thereby demonstrating the benefits of our proposed filter.

Table 8: Tracking performance for different combinations of occlusion models (Occ.) and usage/non-usage of object features (Feat.): MC means and 1 standard deviation (shown in parenthesis, only reported for the main measures). The best result for each sequence is **Bolded**.

Occ.	Feat.	Evaluation with 3D ellipsoid estimates						Evaluation with ground plane estimates					
		FP↓	FN↓	IDs↓	MOTA↑	IDF1↑	OSPA$^{(2)}$↓	FP↓	FN↓	IDs↓	MOTA↑	IDF1↑	OSPA$^{(2)}$↓
IoA	✗	**82**	**292**	34	**89.0(0.43)**	45.5(3.30)	0.92(0.01)	**31**	240	31	91.8(0.44)	45.9(3.30)	0.91(0.01)
	✓	83	328	**27**	88.1(0.69)	**50.9(3.02)**	**0.87(0.02)**	32	277	**24**	91.0(0.51)	**51.9(3.04)**	**0.86(0.02)**
	✓(No Recall)	85	288	34	89.0(0.52)	45.1(2.26)	0.92(0.01)	33	**236**	30	**91.9(0.48)**	45.5(2.26)	0.91(0.01)
LoS	✗	88	301	32	88.6(0.53)	46.6(3.13)	0.91(0.01)	36	249	32	91.5(0.45)	47.0(3.14)	0.91(0.01)
	✓	94	331	**25**	87.9(0.58)	49.0(3.84)	**0.87(0.02)**	41	278	**24**	90.8(0.44)	50.2(3.86)	**0.86(0.02)**
Const.	✗	**55**	330	37	88.6(0.43)	41.7(3.44)	0.93(0.01)	**23**	298	34	90.4(0.43)	42.3(3.34)	0.93(0.01)
	✓	59	356	31	88.0(0.74)	49.7(5.51)	0.88(0.02)	25	322	27	89.9(0.78)	51.2(5.53)	0.87(0.02)

Due to the small difference between the overall tracking performance of the mean-shift clustering (MS) and Gibbs-Sampling (GS) [54] implementations of the adaptive birth model (of Subsection 4.3.1), to distinguish them, we need to examine their OSPA$^{(2)}$ error curves (computed as per Figs. 6 and 7). Fig. 8, indicates that for the WT dataset the MS implementation provides better track initialization than GS with lower OSPA$^{(2)}$ error at the beginning of the scenario. In the CMC dataset, where the area of interest is significantly smaller, the two implementations show nearly identical performance.

5.3.2. Best Hypothesis Approximation

In general, reducing the number of components (hypotheses) decreases the computation time, but at the expense of tracking performance. However, the performance degradation may not be significant in scenarios

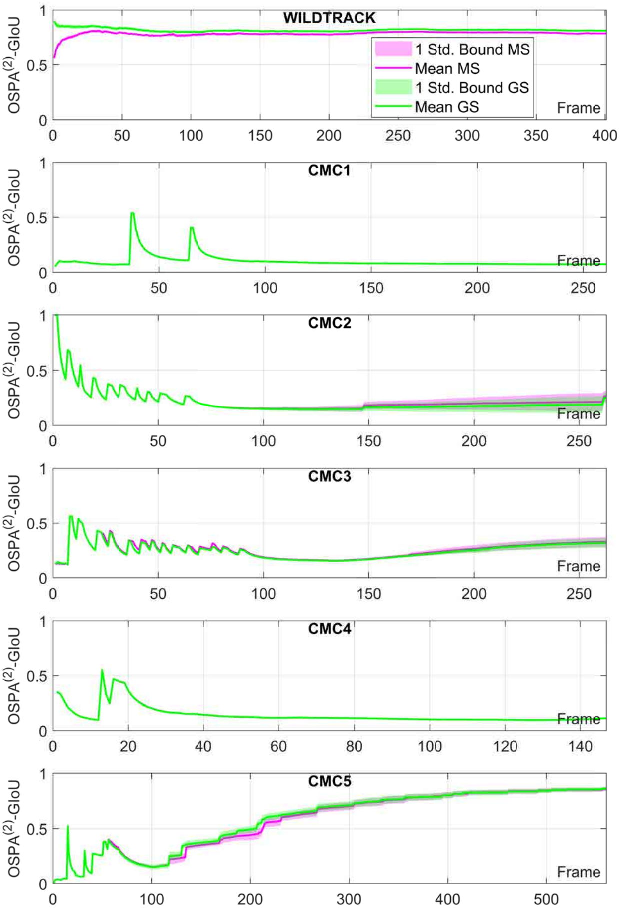

Figure 8: OSPA$^{(2)}$ tracking error (in the *ground plane*) of our filter with the CSTrack detector. MS: mean-shift clustering adaptive birth implementation. Gibbs: Gibbs-sampling adaptive birth implementation.

with a high signal-to-noise ratio (SNR), i.e., high detection probability and low false alarms. In this ablation study, we investigate an extreme case where we only propagate the best hypothesis in the MV-GLMB-AB filter (see Remark 2) and evaluate the performance of this approximate filter on both the CMC and WT datasets.

Tab. 9 presents tracking performance comparison for the MV-GLMB-AB filter and its single-hypothesis approximation. Observe that in the CMC1 and CMC4 sequences, the single-hypothesis MV-GLMB-AB filter is significantly faster than the MV-GLMB-AB filter without significant tracking performance degradation, due to the high SNRs. As expected, in other data sequences where the numbers of miss-detections and false alarms are high, the performance gaps are considerable. Nonetheless, the significant increase in processing speed renders the single-hypothesis MV-GLMB-AB filter suitable for real-time 3D tracking, especially with the continual improvement in detection/segmentation techniques.

Note also from Tab. 9 that the single-hypothesis MV-GLMB-AB filter yields significant increases in ID switches. This is because tracks that are discarded along with the non-optimal hypotheses cannot be recalled

Table 9: Main evaluation criteria for the best-hypothesis approximation and the standard trackers with CSTrack detections. Means and one standard deviations (in parenthesis) are reported for the standard tracker. Evaluation on the CMC dataset is done in 3D, and evaluation on the WT dataset is done in the ground plane. The best result for each sequence is **Bolded**.

Dataset	Tracker	FP↓	FN↓	IDs↓	MOTA↑	IDF1↑	OSPA$^{(2)}$↓	FPS↑
CMC1	Single Hypothesis	**0**	**4**	0	**99.4**	**99.7**	**0.30**	**625.42**
	Multiple Hypotheses	**0**	**4**	0	**99.4(0.00)**	**99.7(0.00)**	**0.3(0.00)**	28.5(0.12)
CMC2	Single Hypothesis	**16**	301	55	82.1	38.5	0.91	**112.34**
	Multiple Hypotheses	18	**36**	**5**	**97.1(1.43)**	**90.2(6.62)**	**0.41(0.03)**	28.5(0.12)
CMC3	Single Hypothesis	78	538	87	75.1	37.6	0.9	**72.58**
	Multiple Hypotheses	**57**	**99**	**15**	**93.9(1.15)**	**78.6(5.03)**	**0.45(0.03)**	7.0(0.33)
CMC4	Single Hypothesis	3	10	1	96.5	86.6	0.5	**208.63**
	Multiple Hypotheses	0	**9**	**0**	**97.5(0.29)**	**98.7(0.16)**	**0.24(0.00)**	4.6(0.11)
CMC5	Single Hypothesis	172	919	149	66.6	11.5	0.99	**71.35**
	Multiple Hypotheses	**83**	**328**	**27**	**88.1(0.69)**	**50.9(3.02)**	**0.87(0.02)**	3.6(0.42)
WT	Single Hypothesis	**1621**	7079	1561	-7.8	7.1	0.99	**16.59**
	Multiple Hypotheses	5040	**2804**	**273**	**14.7(3.37)**	**50.9(2.11)**	**0.79(0.01)**	2.7(0.07)

later when evidence supporting their existence accumulates. As a result, the filter incorrectly initiates new tracks, leading to a high number of ID switches. This can be improved via an ad-hoc scheme that retains significant tracks from discarded hypotheses and recalls them later when there is sufficient evidence supporting their existence. Reducing ID switches in a principled manner requires further investigation.

6. Conclusion

We have exploited recent advancements in 2D detection and multi-view fusion to develop a 3D MV-MOT filter that processes 2D detections from monocular cameras, which avoids expensive 3D object detector training. The proposed MV-MOT filter integrates automatic track initialization, re-identification, occlusion handling, and data association into a single Bayesian filtering framework while at the same time taking advantage of object features to improve efficiency. Performance evaluation on challenging scenarios demonstrated significant improvements of the proposed filter over existing MV-MOT solutions. Ablation studies also show its robustness when camera configurations are changed on-the-fly, and the advantages of the proposed occlusion and adaptive birth models to resolve occlusions and automatically initiates/re-identifies tracks. To the best of our knowledge, the proposed filter is the first to perform track re-identification in 3D from 2D detections. Re-identification could be improved using features that are unique to the objects and time-invariant (or vary slowly with time), which is still an open topic in computer vision.

Acknowledgements

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (No. 2014-3-00077), and Linkage grant (No. LP200301507) funded by the Australian Research Council and Latconnect 60. The authors acknowledge Prof. Ba-Tuong Vo (ba-tuong.vo@curtin.edu.au) and Dr. Changbeom.Shim (changbeom.shim@curtin.edu.au) for their suggestions on the experimental settings and fruitful discussions.

References

[1] G. Thomaidis, M. Tsogas, P. Lytrivis, I. Karaseitanidis, A. J. Amditis, Multiple hypothesis tracking for data association in vehicular networks, Inf. Fusion 14 (4) (2013) 374–383.

[2] S. Blackman, R. Populi, Design and analysis of modern tracking systems(book), Norwood, MA: Artech House, 1999. (1999).

[3] B. Ristic, M. Beard, C. Fantacci, An overview of particle methods for random finite set models, Inf. Fusion 31 (2016) 110–126.

[4] N. Wojke, A. Bewley, D. Paulus, Simple online and realtime tracking with a deep association metric, IEEE Int. Conf. Image Process. (2017) 3645–3649.

[5] E. Bochinski, V. Eiselein, T. Sikora, High-speed tracking-by-detection without using image information, in: IEEE Int. Conf. Adv. Video Signal-Based Surveill., IEEE, 2017, pp. 1–6.

[6] D. Y. Kim, B.-N. Vo, B.-T. Vo, M. Jeon, A labeled random finite set online multi-object tracker for video data, Pattern Recognition 90 (2019) 377–389.

[7] T. T. D. Nguyen, B.-N. Vo, B.-T. Vo, D. Y. Kim, Y. S. Choi, Tracking cells and their lineages via labeled random finite sets, IEEE Trans. Signal Process. 69 (2021) 5611–5626.

[8] C. Liang, Z. Zhang, Y. Lu, X. Zhou, B. Li, X. Ye, J. Zou, Rethinking the competition between detection and reid in multiobject tracking, IEEE Trans. Image Process. 31 (2022) 3182–3196.

[9] Y. Zhang, C. Wang, X. Wang, W. Zeng, W. Liu, FairMOT: On the fairness of detection and re-identification in multiple object tracking, Int. J. Comput. Vis. 129 (11) (2021) 3069–3087.

[10] L. Bridgeman, M. Volino, J.-Y. Guillemaut, A. Hilton, Multi-person 3D pose estimation and tracking in sports, in: IEEE Conf. Comput. Vis. Pattern Recog. Workshops (CVPRW), 2019, pp. 2487–2496. doi:10.1109/CVPRW.2019.00304.

[11] H. Bradler, A. Kretz, R. Mester, Urban traffic surveillance (uts): A fully probabilistic 3D tracking approach based on 2D detections, in: 2021 IEEE Intelligent Vehicles Symposium (IV), 2021, pp. 1198–1205. doi:10.1109/IV48863.2021.9575140.

[12] T. Chavdarova, F. Fleuret, Deep multi-camera people detection, in: IEEE Int. Conf. Mach. learning and Appl. (ICMLA), IEEE, 2017, pp. 848–853.

[13] T. Chavdarova, P. Baqué, S. Bouquet, A. Maksai, C. Jose, T. M. Bagautdinov, L. Lettry, P. V. Fua, L. V. Gool, F. Fleuret, Wildtrack: A multi-camera hd dataset for dense unscripted pedestrian detection, IEEE Conf. Comput. Vis. Pattern Recog. (2018) 5030–5039.

[14] X. Ning, Z. Yu, L. Li, W. Li, P. Tiwari, DILF: Differentiable rendering-based multi-view image–language fusion for zero-shot 3D shape understanding, Inf. Fusion 102 (2024) 102033.

[15] M. Lupión, A. Polo-Rodríguez, J. Medina-Quero, J. F. Sanjuan, P. M. Ortigosa, 3D human pose estimation from multi-view thermal vision sensors, Inf. Fusion 104 (2024) 102154.

[16] P. Baqué, F. Fleuret, P. V. Fua, Deep occlusion reasoning for multi-camera multi-target detection, in: Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 271–279.

[17] J. Ong, B.-T. Vo, B.-N. Vo, D. Y. Kim, S. E. Nordholm, A Bayesian filter for multi-view 3D multi-object tracking with occlusion handling, IEEE Trans. Pattern Anal. Mach. Intell. 44 (2020) 2246–2263.

[18] M. Betke, N. C. Makris, Fast object recognition in noisy images using simulated annealing, in: Proc. IEEE Int. Conf. Comput. Vis., IEEE, 1995, pp. 523–530.

[19] P. A. Viola, M. J. Jones, D. Snow, Detecting pedestrians using patterns of motion and appearance, Int. J. Comput. Vis. 63 (2005) 153–161.

[20] D. G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2004) 91–110.

[21] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Conf. Comput. Vis. Pattern Recog., Vol. 1, IEEE, 2005, pp. 886–893.

[22] R. B. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: IEEE Conf. Comput. Vis. Pattern Recog., 2014, pp. 580–587.

[23] C. L. Zitnick, P. Dollár, Edge boxes: Locating object proposals from edges, in: Eur. Conf. Comput. Vis., Springer, 2014, pp. 391–405.

[24] S. Ren, K. He, R. B. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, Adv. Neural Inf. Process. Syst 28 (2015).

[25] J. Redmon, A. Farhadi, YOLOv3: An incremental improvement, ArXiv abs/1804.02767 (2018).

[26] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, S. Zagoruyko, End-to-end object detection with transformers, in: Eur. Conf. Comput. Vis., Springer, 2020, pp. 213–229.

[27] T.-Y. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft COCO: Common objects in context, in: Eur. Conf. Comput. Vis., Springer, 2014, pp. 740–755.

[28] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, L. Fei-Fei, Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (2015) 211–252.

[29] F. Fleuret, J. Berclaz, R. Lengagne, P. V. Fua, Multicamera people tracking with a probabilistic occupancy map, IEEE Trans. Pattern Anal. Mach. Intell. 30 (2) (2007) 267–282.

[30] P. Peng, Y. Tian, Y. Wang, J. Li, T. Huang, Robust multiple cameras pedestrian detection with multi-view Bayesian network, Pattern Recognition 48 (5) (2015) 1760–1772.

[31] W. Ge, R. T. Collins, Crowd detection with a multiview sampler, in: Eur. Conf. Comput. Vis., Springer, 2010, pp. 324–337.

[32] Y. Hou, L. Zheng, S. Gould, Multiview detection with feature perspective transformation, in: Eur. Conf. Comput. Vis., Springer, 2020, pp. 1–18.

[33] Q. Zhang, W. Lin, A. B. Chan, Cross-view cross-scene multi-view crowd counting, in: IEEE Conf. Comput. Vis. Pattern Recog., 2021, pp. 557–567.

[34] L. Song, J. Wu, M. Yang, Q. Zhang, Y. Li, J. Yuan, Stacked homography transformations for multi-view pedestrian detection, in: Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 6049–6057.

[35] L. Chen, H. Ai, Z. Zhuang, C. Shang, Real-time multiple people tracking with deeply learned candidate selection and person re-identification, in: IEEE Int. Conf. Multimedia Expo, IEEE, 2018, pp. 1–6.

[36] Z. Wang, L. Zheng, Y. Liu, S. Wang, Towards real-time multi-object tracking, in: Eur. Conf. Comput. Vis., Springer, 2020, pp. 107–122.

[37] F. Yu, W. Li, Q. Li, Y. Liu, X. Shi, J. Yan, POI: Multiple object tracking with high performance detection and appearance feature, in: Eur. Conf. Comput. Vis., Springer, 2016, pp. 36–42.

[38] N. Wojke, A. Bewley, Deep cosine metric learning for person re-identification, in: IEEE Winter Conf. Appl. Comput. Vis. (WACV), IEEE, 2018, pp. 748–756. doi:10.1109/WACV.2018.00087.

[39] Y. Wang, K. Kitani, X. Weng, Joint object detection and multi-object tracking with graph neural networks, in: IEEE Int. Conf. Robot. Autom., IEEE, 2021, pp. 13708–13715.

[40] S. M. Khan, M. Shah, A multiview approach to tracking people in crowded scenes using a planar homography constraint, in: Eur. Conf. Comput. Vis., Springer, 2006, pp. 133–146.

[41] R. Eshel, Y. Moses, Homography based multiple camera detection and tracking of people in a dense crowd, in: IEEE Conf. Comput. Vis. Pattern Recog., IEEE, 2008, pp. 1–8.

[42] W. Hu, M. Hu, X. Zhou, T. Tan, J. Lou, S. J. Maybank, Principal axis-based correspondence between multiple cameras for people tracking, IEEE Trans. Pattern Anal. Mach. Intell. 28 (4) (2006) 663–671.

[43] Y. Xu, X. Liu, L. Qin, S.-C. Zhu, Cross-view people tracking by scene-centered spatio-temporal parsing, in: Proc. AAAI Conf. Artif. Intell., Vol. 31, 2017.

[44] Y. Xu, X. Liu, Y. Liu, S.-C. Zhu, Multi-view people tracking via hierarchical trajectory composition, in: IEEE Conf. Comput. Vis. Pattern Recog., 2016, pp. 4256–4265.

[45] T. Zhang, X. Chen, Y. Wang, Y. Wang, H. Zhao, MUTR3D: A multi-camera tracking framework via 3D-to-2D queries, in: IEEE Conf. Comput. Vis. Pattern Recog., 2022, pp. 4537–4546.

[46] Z. Pang, J. Li, P. Tokmakov, D. Chen, S. Zagoruyko, Y.-X. Wang, Standing between past and future: Spatio-temporal modeling for multi-camera 3D multi-object tracking, in: IEEE Conf. Comput. Vis. Pattern Recog., 2023, pp. 17928–17938.

[47] W. Ouyang, H. Zhou, H. Li, Q. Li, J. Yan, X. Wang, Jointly learning deep features, deformable parts, occlusion and classification for pedestrian detection, IEEE Trans. Pattern Anal. Mach. Intell. 40 (8) (2018) 1874–1887.

[48] Y. Ma, Q. Chen, Depth assisted occlusion handling in video object tracking, in: Int. Symp. Vis. Comput., Springer, 2010, pp. 449–460.

[49] D. Stadler, J. Beyerer, Improving multiple pedestrian tracking by track management and occlusion handling, in: IEEE Conf. Comput. Vis. Pattern Recog., 2021, pp. 10958–10967.

[50] X. Yuan, A. Kortylewski, Y. Sun, A. Yuille, Robust instance segmentation through reasoning about multi-object occlusion, in: IEEE Conf. Comput. Vis. Pattern Recog., 2021, pp. 11141–11150.

[51] B.-T. Vo, B.-N. Vo, Labeled random finite sets and multi-object conjugate priors, IEEE Trans. Signal Process. 61 (13) (2013) 3460–3475.

[52] B.-N. Vo, B.-T. Vo, M. Beard, Multi-sensor multi-object tracking with the generalized labeled multi-Bernoulli filter, IEEE Trans. Signal Process. 67 (23) (2019) 5952–5967.

[53] A. Wang, Y. Sun, A. Kortylewski, A. L. Yuille, Robust object detection under occlusion with context-aware composition-alnets, in: IEEE Conf. Comput. Vis. Pattern Recog., 2020, pp. 12645–12654.

[54] A. Trezza, D. J. Bucci, P. K. Varshney, Multi-sensor joint adaptive birth sampler for labeled random finite set tracking, IEEE Trans. Signal Process. 70 (2022) 1010–1025.

[55] Z. Zhang, A flexible new technique for camera calibration, IEEE Trans. Pattern Anal. Mach. Intell. 22 (11) (2000) 1330–1334.

[56] C.-T. Do, T. T. D. Nguyen, H. V. Nguyen, Robust multi-sensor generalized labeled multi-Bernoulli filter, Signal Process. 192 (2022) 108368.

[57] R. Mahler, Statistical multisource-multitarget information fusion, Vol. 685, Artech House Norwood, MA, USA, 2007.

[58] B.-N. Vo, B.-T. Vo, T. T. D. Nguyen, C. Shim, Multi-object estimation beyond the probability hypothesis density filter, ArXiv (2023).

[59] N. Ishtiaq, A. K. Gostar, A. Bab-Hadiashar, R. Hoseinnezhad, Interaction-aware labeled multi-Bernoulli filter, IEEE Trans. Intell. Transp. Syst. (2023).

[60] M. Beard, B.-T. Vo, B.-N. Vo, Bayesian multi-target tracking with merged measurements using labelled random finite sets, IEEE Trans. Signal Process. 63 (6) (2015) 1433–1447. doi:10.1109/TSP.2015.2393843.

[61] B.-N. Vo, B.-T. Vo, D. Phung, Labeled random finite sets and the bayes multi-target tracking filter, IEEE Trans. Signal Process. 62 (24) (2014) 6554–6567.

[62] B.-N. Vo, B.-T. Vo, H. G. Hoang, An efficient implementation of the generalized labeled multi-Bernoulli filter, IEEE Trans. Signal Process. 65 (8) (2017) 1975–1987.

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

[64] K. Bernardin, R. Stiefelhagen, Evaluating multiple object tracking performance: The CLEAR MOT metrics, EURASIP J. on Image and Video Process. 2008 (2008) 1–10.

[65] M. Beard, B.-T. Vo, B.-N. Vo, A solution for large-scale multi-object tracking, IEEE Trans. Signal Process. 68 (2020) 2754–2769.

[66] T. T. D. Nguyen, H. Rezatofighi, B.-N. Vo, B.-T. Vo, S. Savarese, I. Reid, How trustworthy are the existing performance evaluations for basic vision tasks?, IEEE Trans. Pattern Anal. Mach. Intell. (2022).

[67] R. Qiu, M. Xu, Y. Yan, J. S. Smith, X. Yang, 3D random occlusion and multi-layer projection for deep multi-camera pedestrian localization, in: Eur. Conf. Comput. Vis., 2022.