IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 31, 2018, accepted February 28, 2018, date of publication March 6, 2018, date of current version March 28, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2812794

Data Fusion-Based Multi-Object Tracking for Unconstrained Visual Sensor Networks

XIAOYAN JIANG[1], ZHIJUN FANG[1], (Senior Member, IEEE), NEAL N. XIONG[2], (Senior Member, IEEE), YONGBIN GAO[1], BO HUANG[1], JUAN ZHANG[1], LEI YU[1], AND PATRICK HARRINGTON[2]

[1]School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China
[2]Department of Mathematics and Computer Science, Northeastern State University, OK 74464, USA

Corresponding author: Zhijun Fang (zjfang@sues.edu.cn) and Neal N. Xiong (xiong31@nsuok.edu)

This work was supported in part by the National Natural Science Foundation of China under Grant 61702322 and Grant 61772328; in part by the International Cooperation Project of Jiangxi Province under Grant 20133BDH118011; in part by the Young Teacher Development Program of Shanghai under Grant ZZGCD16008.

ABSTRACT Camera node perception capability is one of the crucial issues for visual sensor networks, which belongs to the field of Internet of Things. Multi-object tracking is an important feature in analyzing object trajectories across multiple cameras, thus allowing synthesis of data and security analysis of images in various scenarios. Despite intensive research in the last decades, it remains challenging for tracking systems to perform in real-world situations. We therefore focus on key issues of multi-object state estimation for unconstrained multi-camera systems, e.g., data fusion of multiple sensors and data association. Unlike previous work that rely on camera network topology inference, we construct a graph from 2-D observations of all camera pairs without any assumption of network configuration. We apply the shortest path algorithm to the graph to find fused 3-D observation groups. Our approach is thus able to reject false positive reconstructions automatically, and also minimize computational complexity to guarantee feasible data fusion. Particle filters are applied as the 3-D tracker to form tracklets that integrate local features. These tracklets are modeled by a graph and linked into full tracks incorporating global spatial-temporal features. Experiments on the real-world PETS2009 S2/L1 sequence show the accuracy of our approach. Analyses of the different components of our approach provide meaningful insights for object tracking using multiple cameras. Evidence is provided for selecting the best view for a visual sensor network.

INDEX TERMS Data fusion, graph theory, Internet of Things, particle filters, visual sensor networks.

I. INTRODUCTION

In today's society, vehicles, phones, sensors, and other objects are connected across networks to provide intelligent services with minimal human intervention. These networks are known as Internet of Things. Due to the widely deployed cameras over the world, abundant research has been done regarding meaningful data and information extraction from the extensive videos produced by visual sensor networks (VSNs). VSNs are also known as camera sensor networks [1], [2]. Due to the directional property of cameras in the network [3], some research focuses on proposing optimization-based algorithms to obtain optimal configuration of the network. This includes, for example, optimal orientation [4], optimal location, and full-view coverage problems [1], [5]. With the exception of network transmission performance related to 5G technology, the capability of mobile computing and visual perception by networked nodes is an important factor to judge optimal network configuration. An important and challenging issue is accurate tracking of multiple targets in the camera network to obtain an intelligent VSN system. Trackers provide significant hints, *i.e.*, object locations and identities, for the sensors to continuously recognize the action and intention of objects in VSNs.

Compared with static camera networks, current VSNs have high network configuration flexibility that includes camera locations, viewing angles, and sensor types that can be used. This flexibility leads to greater difficulty in tracking multiple objects across cameras, especially with regard to illumination, object occlusion, and different sensor parameters. Due to improvements in object detection algorithms with regard to accuracy and computational feasibility, tracking-by-detection has become a popular framework [6], [7] to solve the multi-object tracking problem. In this approach, detectors obtain object hypotheses that normally contain false positives and false negatives. The primary goal after applying the algorithms is to maintain correct data association for objects' individual trajectories. However, difficulties are encountered for single-view tracking when multiple objects are occluded or when there are ambiguities in distinguishing different targets. In such situations, trackers can benefit from data fusion of other views by using multiple cameras. Since it is safe to assume that different objects do not occupy the same position in 3D space, conducting tracking on 3D detections potentially solves the problem of 2D occlusions. As indicated by previous works [8], [9], there are mainly two distinct frameworks to track multiple objects using data obtained from multiple views, depending on how the 3D data is integrated. The two frameworks are described briefly in the following:

A. ACROSS-TIME FOLLOWED BY ACROSS-VIEW ASSOCIATION

It is also called the *tracking-reconstruction* approach [8]. 2D tracks are generated in single cameras using 2D detections of each view independently. Afterwards, 2D trajectories are associated across distinct cameras and the matched 2D trajectories of different views are used to form 3D object trajectories by reconstructing positions at each time step.

B. ACROSS-VIEW FOLLOWED BY ACROSS-TIME ASSOCIATION

It is also called *reconstruction-tracking* approach [8], [9]. Here, 2D detections of multiple views are first reconstructed into 3D measurements. Detection correspondences are found by judging appearance similarities and the distances to corresponding epipolar lines [10]. Subsequently, data association is directly conducted on 3D detections to form 3D object trajectories. The strategy overcomes early-stage occlusions that can occur in a 2D image space.

In our paper, we focus on analyzing and improving the performance of *Across-View* Followed by *Across-Time* Association approach. Considering computational complexity, center points of the rectangles instead of object point clouds are used as the original 2D detection for creating 3D reconstructions. Due to inaccurate localization of the detector and calibration errors, reconstructions of the same objects in the 3D world coordinate system from multiple camera views can not be localized accurately. Moreover, matching and localizing corresponding objects in two separate views is challenging because of large differences between camera viewing angles, lighting conditions, and 2D occlusions. As a result, the *ghost effect* [8], [9] occurs, which leads to 3D false positive and missing detections. Fig. 1 shows one example of the *ghost effect* using the PETS dataset [11]. As multiple detections in two cameras are similar in appearance and spatially close to corresponding epipolar lines, false positive 3D detections are reconstructed. In order to overcome this problem, sparse matching for reconstruction was formulated as a linear assignment problem in [12]. However, when there are more than three cameras, the problem is NP-hard [13]. To create more accurate 3D measurements from multiple views, a novel data fusion algorithm is presented which has linear computational complexity. It works on 3D detections directly and combines evidence from 2D images to create 3D measurements with smaller false positive and false negative rates.

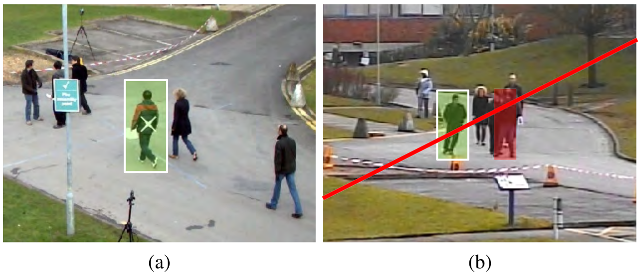


FIGURE 1. An example of false corresponding pairs of detections between two views in the PETS/S2.L1 dataset [11]. Multiple candidates (rectangles in (b)) are near the epipolar line (red line in (b)) for the query object (white rectangle in (a)). Both images have the same frame index in the sequence.

We propose a multi-object tracking framework using multi-camera systems without the necessity of tuning parameters from the ground truth of the sequence [14]. The algorithm does not rely on camera network topology inference and the type of sensors are flexible. Thus, the multi-camera system is unconstrained. Our work improves several aspects of other recent works [15], [16] and optimizes target trajectories both locally and globally. The contributions of the paper are: (1) We present a novel graph-based data fusion algorithm to estimate accurate 3D observations of multiple views, which assembles both 2D and 3D features including multi-view geometry information. (2) In contrast with popular tracking algorithms, such as network flow-based [17] and graph-based [16], particle filters are used for extracting tracklets, in which motion information is easily incorporated. (3) A comprehensive analysis of different components of the approach is presented. The influence of detection performance on tracking-by-detection is studied.

The paper is arranged as follows. We discuss related work in Section II. Section III presents the whole approach including graph-based data fusion, particle filter-based tracklet generation, and graph-based tracklet linking. Experiments and analysis of parameters are shown in Section IV. Section V summarizes the paper and proposes possible future work.

II. RELATED WORK

Most tracking approaches [18], [19] use centralized systems by fusing the information from multiple views in a joint manner at each time step. The key point then is an accurate coupling model that captures useful evidence from all the cameras, since individual views can not contribute to the tracking system all the time in the same way. For real-time required applications when the number of cameras increases, an efficient data fusion applied on a powerful computer is in demand. In contrast, other approaches use distributed systems that perform differently [18] from centralized systems. Object tracking is first conducted independently on a 2D image space, and afterwards tracks from individual cameras are combined globally into 3D trajectories.

Reconstruction of objects seen from several views is also known as a sparse stereo matching problem [12]. However, matching objects across views are difficult because of changing view angles, distinct object poses, illumination, and occlusion. When the number of cameras is more than three, the matching problem is NP-hard [13]. As a result, feature based matching [20] is not suitable for large-distance camera network scenarios. To reconstruct data, an applicable and simplified usage is to reconstruct data on a specific plane from individual views. Khan and Shah [21] proposed a novel planar homography constraint to resolve occlusions from multiple observations to determine the locations of the feet of the corresponding people walking across the ground. The work of [22] reconstructed the top-view of the ground plane, mapped the vertical axes of people in each view to the top-view, to intersect at single points that are assumed to be the locations of the persons on the ground. Fleuret *et al.* [23] combined dynamic programming with the estimation of occupancy probabilities on the ground plane at each time step from all view images that are background subtracted.

Another approach uses multi-person tracking scenarios combined with reconstruction in order to solve one objective function. Leal-Taixè *et al.* [12] built graphs to solve a combinatorial optimization problem of reconstruction and tracking, taking all available evidence into account. In [24], tracking in single cameras and estimation of 3D trajectories are solved simultaneously, by searching optimal sub-hypergraphs on space-time-view hyper-graphs. Liem and Gavrila [25] further integrated appearance information to the objective function. Later on, Hofmann *et al.* [14] followed the work. In it, they did jointly compacted reconstruction and tracking into one maximum a posterior estimation issue. For this purpose, an assessment of the detection performance, *e.g.* false positive and missing detections, is required in the tracking framework.

Since data association is the key problem for detection-based multi-object tracking, there are numerous works presenting possible solutions [6], [17]. Classic data association approaches include *e.g.*Multiple Hypotheses Tracking (MHT) [26], and Joint Probability Data Association (JPDA) [27]. Possible origins of target measurements in MHT are accounted by a set of data association hypotheses. For measurements per time unit, probabilities that the measurement belongs to previous tracks, pertains to a new target, or is a false are calculated. After several time steps as measurements are received, probabilities of joint hypotheses are computed recursively. Due to complexity, the analysis is limited to only few such steps [6]. In comparison to the calculation of posterior probabilities for single measurements, JPDA considers independence between objects and computes joint conditional probabilities for data association. However, the computational complexity grows exponentially with the number of targets.

Breitenstein *et al.* [6] proposed a greedy data association method to match detections with particle filters in consecutive frames. In [28], bipartite graphs were employed for tracking, with judgments based on likelihood functions embedded into the weight functions. In contrast to these methods that search for a local optimum, global optimization schemes aim to model and investigate over the entire sequence, such as network flow-based [17], [29] shortest path algorithm, global energy function minimization [30], and parameter optimization for the min-cost flow of multi-object tracking problem [31].

Due to noisy detections, however, direct linking among detections is error-prone in difficult situations, such as occlusion. As for the case of occlusion, multiple objects tend to merge and split afterwards, which makes it difficult for the trackers to maintain the correct identities of the objects. In addition, as objects cross each other, the same problem is encountered again. Recently, many approaches separated the tracking process into several stages [7], [15], [16], [32] in order to produce more reliable tracklets in each step and link into longer tracks. Huang *et al.* [7] proposed a three-level hierarchical data association approach. At the low level, affinity constraints was used to link detection responses into tracklets. The Hungarian algorithm was employed to link these tracklets into more reliable tracklets in the second level. Finally, entries, exits and scene occlusions were estimated to refine the final trajectories. Xing *et al.* [15] utilized particle filters for local tracklet extraction and generated global tracklets within a sliding window thereafter. Jiang *et al.* [16] presented a two-stage graph based multi-person tracking approach. Tracklets and tracks were produced by traversing the nodes of the shortest paths in two individual graphs.

We follow the framework of tracklet extraction and tracklet linking. Our approach differs from previous work. For example, consider a person walking on a flat plane across the ground. Assuming that calibration parameters are provided, for each time step we extract a set of 2D points that are composed by the middle points on the bottom margins of the 2D rectangle detections in each view. These points are projected onto the ground plane, and tracking is then performed based on the projected detections on the ground plane. Due to calibration errors, multiple detections on ground plane in one time step may arise from single objects. Therefore, we propose a fusion method based on a graph model to separate them into groups. These grouped detection hypotheses are used to initialize and update the particle filter as a local tracker. Tracklets are subsequently extracted from local trackers performed over the whole sequence. Tracklets are linked by a directed acyclic graph using global temporal and spatial features.

Since the most similar work to us is [16], we list the attributes of this work as follows: 1) Original reconstructions from more than two cameras are fused to obtain more accurate

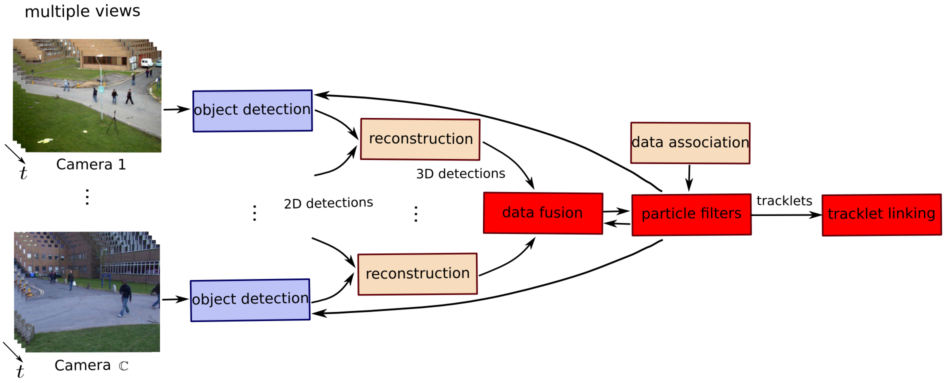


FIGURE 2. The *across-view* data association (reconstruction and data fusion) followed by the *across-time* data association (data association, particle filters, and tracklet linking) framework.

observations; 2) Motion information is incorporated into the extraction of tracklets by particle filters; 3) 2D evidence of all views is fused by a weighted graph.

III. THE ALGORITHM

A. OVERVIEW

Assume that we have the access to 2D detections of each frame of all camera videos. The original set of 3D reconstructions are triangulated from 2D detections from all pairs of cameras using the epipolar geometry restriction. Fig. 2 shows the framework of the proposed approach, which contains three main stages: graph-based data fusion, particle filter-based tracklet extraction, and graph-based tracklet linking. Given the original set of 3D reconstructions, the goal of the data fusion module is to create more accurate 3D observations. Additionally, the tracked positions obtained by particle filters at the current time step are used in the fusion step at the next time step. This makes the fusion process guided by the previously tracked positions, which are temporally more reliable than original reconstructions. Particle filters are initialized and updated by temporally connected 3D observations created by the greedy data association algorithm, where motion models are also adopted for object state prediction. After analyzing the whole sequence through particle filters, tracklets are generated. Finally, a graph is built to globally link tracklets into tracks.

B. DATA FUSION FROM SEVERAL CAMERAS

As indicated before, original 3D reconstructions from multiple views are ambiguous, *e.g.* duplicated detections for single objects, which results in high false positive rates. To generate more accurate measurements, a graph is constructed to fuse 3D reconstructions at each time step. Reconstructions at the current time step and tracked object positions from the last time step are represented as nodes in the graph. Accordingly, the shortest path algorithm is applied to the graph to find node groups. A group refers to a set of 3D reconstructions belonging to an identical object. As a result, detections in the same groups are fused in order to generate single reconstruction hypotheses for individual objects. An illustration of the fusion process is shown in Fig. 3, where 3D detections are observed from the top view for a better explanation.

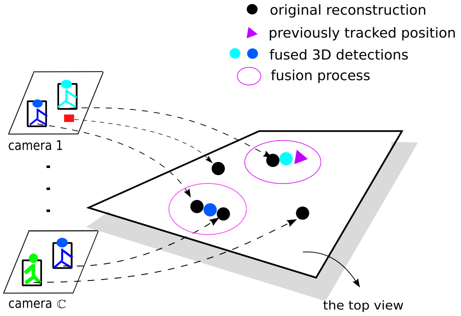


FIGURE 3. The fusion of detections for people which are shown by the top view. Reconstructions at the current time step and tracked positions at the previous time step are fused to obtain more accurate object observations. Color indicates object identity.

We denote the set of detection candidates at time t as $\mathbf{X}_t = \mathbf{D}_t \cup \mathbf{T}_{t-1}$, which is composed of the current set of original reconstructions $\mathbf{D}_t$ and the set of previously tracked positions $\mathbf{T}_{t-1}$ derived from particle filters. Each $\mathbf{x}_i^t \in \mathbf{X}_t \subseteq \mathbb{R}^3$ has a corresponding 2D detection in one view. At each time step, we construct a weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$, where each node $v_i \in \mathcal{V}$ represents one location $\mathbf{x}_i$ and w is a weighted function of edges, *i.e.* $w : \mathcal{E} \rightarrow \mathbb{R}$. Any two nodes v_i, $v_j \in \mathcal{V}, i \neq j$ are connected by an undirected edge $e_{i,j} = \{v_i, v_j\}, e_{i,j} \in \mathcal{E}$, where i, j are indices. The corresponding weighting function $w_{i,j}$ is defined as follows:

$$w_{i,j} = w(v_i, v_j) \tag{1}$$

$$= \begin{cases} log\left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{\alpha}\right) & \|\mathbf{x}_i - \mathbf{x}_j\| < \alpha \ \& \\ & f_{bp}(\mathbf{x}_i) = f_{bp}(\mathbf{x}_j) \\ \infty & \text{otherwise.} \end{cases} \tag{2}$$

The function $\|\cdot\|$ calculates the Euclidean distance between two points. The threshold α is the maximally allowed Euclidean distance between two points that can be connected, since points that are far away unlikely belong to identical objects. This constraint of distance thresholding is intuitive, and the function gives lower weights to the two nodes that are spatially close. The projection function in the second constraint is defined as $f_{bp}(\cdot) = \{f_{bp}^{1}(\cdot), \dots, f_{bp}^{\mathbb{C}}(\cdot)\}$. For $c \in \{1, \dots, \mathbb{C}\}$, $f_{bp}^{c}(\mathbf{x}_i) = \{x_c^i, y_c^i, sx_c^i, sy_c^i\}$ only holds when the projection point of $\mathbf{x}_i$ into view c is inside the rectangle represented as $\{x_c^i, y_c^i, sx_c^i, sy_c^i\}$. (x_c^i, y_c^i) is the top left point and sx_c^i, sy_c^i represents the width and height of the rectangle, respectively. Therefore, the equal sign in the second constraint of Equ. 2 is valid only when two hypothesis locations $\mathbf{x}_i$ and $\mathbf{x}_j$ are projected into the same 2D detection rectangles in all identical views. That is to say, it provides evidence in 2D image space that the two connected nodes likely represent identical objects. In practice, $\|\mathbf{x}_i - \mathbf{x}_j\|$ is nonzero due to calibration errors and detection errors in the image. When $w_{i,j}$ is finite, we get $log\left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{\alpha}\right) < 0$ since $\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{\alpha} \in (0, 1)$.

A virtual source node v_{source} and a virtual sink node v_{sink} are defined to start and terminate paths in the graph, respectively. The weights for connecting an arbitrary node $v_i \in \mathcal{V}$ with the two virtual nodes are assigned as:

$$w(v_{\text{source}}, v_i) = w_{\text{comp}}, \tag{3}$$

$$w(v_i, v_{\text{sink}}) = w_{\text{comp}}, \tag{4}$$

where $w_{\text{comp}} \in \mathbb{R}^+$ can be any positive value. We iteratively employ the Bellman-Ford shortest path algorithm [33] to obtain a set of shortest paths in the graph.

Define a calculated shortest path as $(v_{\text{source}}, v_1, \dots, v_m, v_{\text{sink}})$, where $m + 2$ is the total number of nodes in this path. The nodes $v_1, \dots, v_m$ between v_{source} and v_{sink} represent a group $\{\mathbf{x}_1, \dots, \mathbf{x}_m\}$ where $\forall \mathbf{x}_i, i = \{1, \dots, m\}$ belongs to the same object. The corresponding new detection hypotheses $\mathbf{X}'_t \in \mathbb{R}^3$ are estimated from the groups, in which each hypothesis $\mathbf{x}' \in \mathbf{X}'_t$ is the center of one group:

$$\mathbf{x}' = \frac{1}{m} \sum_i \mathbf{x}_i. \tag{5}$$

By fusion of data from multiple cameras, duplicated reconstructions are merged into single ones, and false positive detections in certain views are removed. The computation complexity is nearly linear to the number of nodes regardless the number of cameras, while the multi-dimensional assignment for multi-sensor data fusion proposed in [13] is NP-hard when there are more than three cameras. It can achieve a real-time performance in practice. Note that objects detected by only one camera produce the "new" missing reconstructions induced in this process, since at least two nodes are required to form a path in the fusion graph.

C. ESTIMATION WITH 3D PARTICLE FILTERS

The following section presents a 3D particle filter-based tracking approach known as the set of 3D detections. Given $\Omega = (\mathbf{X}'_0, \cdots, \mathbf{X}'_t)$ as 3D object detections until time t, particle filters are generated. For each unassigned 3D detection, a new particle filter is initialized and assigned. The state of a particle $\mathbf{s} = \{\mathbf{x}, \pi\}$ is six dimensional, *i.e.* $\mathbf{x} = \{x, y, z, \Delta x, \Delta y, \Delta z\}$ which represents the 3D position (x, y, z) of the object and the corresponding framewise motion $(\Delta x, \Delta y, \Delta z)$. The condensation algorithm [34] is employed and we keep a constant number N of particles over the entire time span for each track. The importance factor π_t^n for the n^{th} particle of a track at time t can be written as $\pi_t^n \propto p(\mathbf{z}_t | \mathbf{x}_t = \mathbf{s}_t^{(n)})$, which is proportional to the conditional likelihood of the measurement $\mathbf{z}_t$ at time t given the state of the particle $\mathbf{s}_t^{(n)}$. It is defined as follows:

$$p(\mathbf{z}_t | \mathbf{x}_t = \mathbf{s}_t^{(n)}) = \left(d_{\text{spat}}^{3D} \cdot d_{\text{spat}}^{2D}\right)^{-1}, \tag{6}$$

where

$$d_{\text{spat}}^{3D} = \|pos(\mathbf{z}_t) - pos(\mathbf{s}_t^{(n)})\|, \tag{7}$$

$$d_{\text{spat}}^{2D} = \sum_c \|f_{bp}^c(\mathbf{z}_t) - f_{bp}^c(\mathbf{s}_t^{(n)})\|. \tag{8}$$

The function $pos(\cdot)$ obtains the 3D position of the particle $\mathbf{s}_t^{(n)}$ or the measurement $\mathbf{z}_t$. The more distant two positions are, the smaller the value π_t^n that is assigned. Note that not only 3D spatial distances, but also projected 2D distances are used. As indicated in Equ. 8, distances between the projections of $\mathbf{s}_t^{(n)}$ and $\mathbf{z}_t$ in each camera are computed. The definition of the function f_{bp} can be found in Equ. 2. The item $\|f_{bp}^c(\mathbf{z}_t) - f_{bp}^c(\mathbf{s}_t^{(n)})\|$ calculates the Euclidean distance between the center points of the two projected rectangles. Moreover, other features can also be used to obtain the distance between the two rectangles.

A constant motion model as in [6] is utilized to propagate the particles:

$$(\Delta x, \Delta y, \Delta z)_t = (\Delta x, \Delta y, \Delta z)_{t-1} + \epsilon_{(\Delta x, \Delta y, \Delta z)}, \tag{9}$$

where $\epsilon_{(\Delta x, \Delta y, \Delta z)}$ is drawn from a zero-mean normal distribution independently. The variance of the distribution keeps accordance with the movement of the target. The motion $(\Delta x, \Delta y, \Delta z)_{t-1}$ is set to be equal to the difference of associated detections at the previous time step. Therefore, the motion prediction model is in accordance with the motion of associated detections. However, a simple constant velocity has been proven to be powerful enough to model the motion of objects in many real scenarios [6].

1) INITIALIZATION

Every single detection $\mathbf{x}'_t \in \mathbf{X}'_t$ at time t is considered to trigger a new track if it is not associated with any existing track, *e.g.* τ'. The binary decision function for initializing a track by $\mathbf{x}'_t$ is defined as follows:

$$I(\mathbf{x}'_t) = \begin{cases} 1 & \forall \tau', \ \|\mathbf{x}'_t - \tau'_{t-1}\| > \theta \\ 0 & \text{otherwise}, \end{cases} \tag{10}$$

where τ'_{t-1} the the position of τ' at $t - 1$. The parameter θ is the maximally allowed motion, which can be set heuristically according to the application. Initial particles are normally distributed around $\mathbf{x}'_t$ with $\sigma^2 = 1$ as variance. Original particles are assigned with the same normalized weight, *i.e.* $\frac{1}{N}$. The initial motion of the tracker is set to be the the same as the movement of associated detections. After initialization, particles are propagated to new states according to the motion model in Equ. 9 at each time step.

2) TERMINATION

In order to obtain accurate object trajectories in ambiguous cases, tracks should be terminated and new ones re-initialized. One case is when the target of a track has not been updated for a successive number of frames. This happens either because of large changes of the object compared with the previous frames, or due to missing detections in the system. When targets are too close to each other and occlusions occur in all the cameras, the fusion stage groups them into single objects. Hence, only single detections are estimated for several objects, which results in missing detections. In this case, the updating of particle filters is often not reliable.

3) GREEDY DATA ASSOCIATION IN 3D

A greedy data association approach similar to the method presented in [35] is applied to find detection and track assignments at each time step. Instead of using 2D data, 3D detections are utilized. Once the associated detection is assigned, it plays a crucial role in guiding the corresponding track.

The likelihood function for weighting the connection of $\mathbf{x}'$ and τ' can be defined as follows:

$$L_{3\mathrm{D}}(\mathbf{x}', \tau') = L'_{\mathrm{pos}}(\mathbf{x}', \tau'), \tag{11}$$

where $L'_{\mathrm{pos}} = \|\mathbf{x}' - \tau'\|$ and $L'_{\mathrm{pos}} < \theta$. The two items, *i.e.* $\mathbf{x}'$ and τ', can be projected to images in different views to obtain evidence. Note that useful features, *e.g.* histogram similarity and 2D spatial distances, give hints to $L_{3\mathrm{D}}(\cdot)$ as well. The state of the target represented by τ' is then updated as $\mathbf{x}'$.

4) OBSERVATION

The final state of a track is estimated by the maximum of the modeled posterior distribution:

$$\boldsymbol{x}_t = \underset{w_t^{[n]},\ n=1\ldots N}{\arg\max}\ \boldsymbol{x}_t^{[n]}, \tag{12}$$

where $\boldsymbol{x}_t$ is the resulting state of the track at time t and $w_t^{[n]}$ is the corresponding weight assigned to the particle $\boldsymbol{x}_t^{[n]}$.

Occlusion Reasoning: When there are occlusions, updates to the particle filter are often not reliable. Therefore, we terminate a tracker when there is ambiguity. If two trackers are associated to the same detection at the same time, they are both terminated. When the target of a tracker has not been updated for successive N frames, it is also terminated. Additionally, multiple tracklets that are close to each other for a certain number of frames are merged to single ones as in [16]. Due to these rules, the tracklets are shorter and more reliable compared to methods where local tracklets are unchangeable once wrongly formed [16]. Once reliable tracklets are acquired, we can link them together into longer tracks.

D. TRACKLET LINKING

Since the number of objects and the length of an object's existence over the sequence are unknown a priori, the linking of tracklets should be flexible. We construct a tracklet graph $\mathcal{G}' = \left(\mathcal{V}', \mathcal{E}', w'\right)$, where each vertex $v'_k \in \mathcal{V}'$ represents a tracklet $\tau_k = \left(\wp_k, t_{\tau_k,0}, t_{\tau_k,1}\right)$ from frame $t_{\tau_k,0}$ to frame $t_{\tau_k,1}$. $\wp_k$ stores the sequential positions, which is similar to the work of [16]. Directed edges $e'_{k,l} = (v'_k, v'_l) \in \mathcal{E}'$ connect vertices, which results in a graph structure. The weight function $w'_{k,l}$ is defined as follows:

$$w'_{k,l} = w'(v'_k, v'_l) \tag{13}$$

$$= \begin{cases} -log(d^{k,l}_{\mathrm{spat}}) - log(d^{k,l}_{\mathrm{temp}}) & t_{\tau_{l,0}} - t_{\tau_{k,1}} > 0\ \& \\ & t_{\tau_{l,0}} - t_{\tau_{k,1}} < \beta \\ \infty & \text{otherwise,} \end{cases} \tag{14}$$

$$d^{k,l}_{\mathrm{spat}} = \|\wp_k(t_{\tau_{k,1}}) - \wp_l(t_{\tau_{l,0}})\|, \tag{15}$$

$$d^{k,l}_{\mathrm{temp}} = t_{\tau_{l,0}} - t_{\tau_{k,1}}, \tag{16}$$

where d_{spat}, d_{temp} are spatial and temporal distance of the considered vertices, respectively. Temporal threshold β is the maximally allowed number of frames between two tracklets.

Similar to the data fusion stage, we define a virtual source node v'_{source} and virtual sink node v'_{sink} and configure an equal positive weight connecting each vertex $v'_k \in \mathcal{V}'$ from v'_{source} and to v'_{sink}:

$$w\prime(v'_{\mathrm{source}}, v\prime_k) = w\prime(v\prime_k, v'_{\mathrm{sink}}) = d\prime_{\mathrm{penalty}} \in \mathbb{R}^+. \tag{17}$$

Moreover, if a tracklet starts in the first frame of the sequence, it is assumed that the represented object appears in an entrance. If a tracklet terminates in the last frame of the video, it is assumed that the represented object disappears in an exit. Thus, a smaller positive weight can be assigned to the tracklets that start or terminate tracks:

$$w\prime_{\mathrm{source}}(v\prime_k) = d\prime\prime_{\mathrm{penalty}}, \tag{18}$$

$$w\prime_{\mathrm{sink}}(v\prime_k) = d\prime\prime_{\mathrm{penalty}}, \tag{19}$$

$$d\prime\prime_{\mathrm{penalty}} \in \mathbb{R}^+, \quad d\prime\prime_{\mathrm{penalty}} < d\prime_{\mathrm{penalty}}. \tag{20}$$

We again employ the Bellman-Ford shortest path algorithm [33]. The calculated paths are traversed to form final trajectories. Consequently, all available connections of tracklets are encouraged to be globally linked with low weights. The pseudo code for the whole algorithm is shown in Algorithm 1.

IV. EXPERIMENTS

In the following section, the performance of each component of the approach is evaluated and compared with state-of-the-art algorithms. Note that no preprocessing is implemented, *e.g.* learning the foreground or training the detector for the specific scenario to be analyzed. No assumptions are made, for example in [7], entries, exits, and occluders on the ground plane are modeled.

Algorithm 1 Multi-Object Tracking for VSN

```
Input: a number of $\mathbb{C}$ sensor videos
Output: 3D trajectories of multiple objects
for each time $t \in [1, \mathbb{T}]$ do
    for all camera $c \in [1, \mathbb{C}]$ do
        2D detection
    end for
    $\mathbf{D}_t \leftarrow$ 3D reconstructions
    $\mathbf{X}_t = \mathbf{D}_t \cup \mathbf{T}_{t-1}$
    construct an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$, where
    $v_i \in \mathcal{V}$ represents 3D reconstruction $\mathbf{x}_i \in \mathbf{X}_t$ and
    $e_{i,j} = \{v_i, v_j\}, e_{i,j} \in \mathcal{E}$ with corresponding $w_{i,j}$
    compute $w_{i,j}$ according to Equ. 2
    find shortest paths in $\mathcal{G}$
    estimate new hypotheses $\mathbf{X}'_t \in \mathbb{R}^3$
    Tracklets $\mathcal{T} \leftarrow$ 3D particle filters estimation
    construct a directed graph $\mathcal{G}' = (\mathcal{V}', \mathcal{E}', w')$, where
    $v'_k \in \mathcal{V}'$ represents a tracklet $\tau_k \in \mathcal{T}$ and $e'_{k,l} =$
    $(v'_k, v'_l) \in \mathcal{E}'$
    compute $w'_{k,l}$ according to Equ. 14
    Tracks $\leftarrow$ find shortest paths in $\mathcal{G}'$
end for
```

A. EXPERIMENTAL SETUP

1) EVALUATION METRICS

A fair comparison of the quantitative evaluation for multi-object tracking is challenging [36]. For our evaluation metric, we employ the widely accepted CLEAR MOT [37], which has become the default standard for evaluating multiple object tracking algorithms.

a: TRACKING EVALUATION

Two important metrics are included in CLEAR MOT, namely MOTP and MOTA. The former metric allows us to independently assess the precision of the tracker regardless of whether the correct object identity matches. The latter metric provides information about misses, mismatches (ID switch), and false positives of the track. For details, please refer to [37].

b: DETECTION EVALUATION

In the case of tracking-by-detection approaches, the performance of the tracker heavily relies on accurate detections. As detectors can be easily replaced, an evaluation of the detectors themselves are useful to estimate influence on the final result. Consequently, an evaluation metric similar to the ones mentioned above is used, namely MODP and MODA as proposed by [38]. MODP is defined the same way as MOTP by evaluating the detector precision, and only considering those matched detections with ground truth data. In contrast, MODA measures the rate of missing and false positive detections, without considering mismatches as MOTA does.

2) DATASET

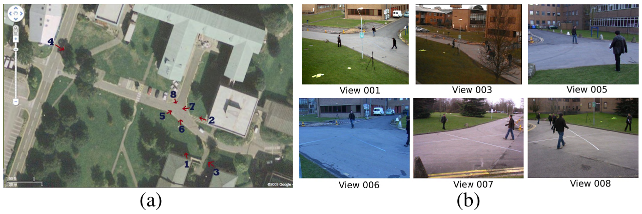


FIGURE 4. **(a) Camera locations for the PETS dataset. (b) Sample images in the corresponding PETS/S2.L1 videos of six available views [11].**

Few public datasets are suitable for multi-person tracking in multi-camera systems. In recent years, the PETS/S2 dataset [11], [38] has become a benchmark in the field of multi-person tracking. Since it contains videos captured by multiple cameras, making this a challenging problem to solve, we implement and test our approach on this dataset. Many state-of-the-art algorithms reported their evaluation scores on this dataset, which enables us to make a direct comparison with our own algorithm. The benchmark dataset for multi-object tracking consists of three different sets of videos, *i.e.* PETS/S2.L1, PETS/S2.L2, and PETS/S2.L3, involving different numbers of cameras. In total eight cameras are available, for which the frame resolution for the first four is 768 × 576 pixels and the rest 720 × 576 pixels. The dataset organizer provided Google maps showing the camera locations, as shown in Fig. 4a. The framerate for all the cameras is $\sim 7 f/s$. In all our experiments, we used the provided calibration parameters for the cameras. Since PETS/S2.L1 has been used as a *de facto* standard database for tracking multiple persons, its use for tracking assessment is important. The goal of this database is to accurately localize individual people over the sequence, including the bounding boxes and IDs. Different 2D occlusion types and human movement patterns are included. Seven calibrated cameras were adopted to observe a large area resulting in multi-view sequences, each of which were recorded 795 frames. We did not use the camera that has synchronization problems against others. Fig. 4b shows sample pictures captured by six of the available cameras at the same time in the PETS/S2.L1 database. Furthermore, the ground plane is $Z = 0$ in the world coordinate system. In our experiments, detections on the ground plane are used as input to the 3D tracker, *i.e.* AViewF. Note that our approaches do not have the limitation that people in the video must walk on the ground plane, but our work is applicable to unconstrained 3D world coordinate system.

3) GROUND TRUTH

Since the first view in the PETS/S2 dataset, *i.e.* view 001, has the broadest field of view compared with other cameras in the system, and many work in the literature [6], [39] track and evaluate only based on this view, we also evaluate our tracking performance using only this view. Anton Andriyenko [40] provided two different ground truth tracks: a cropped one and a complete one, both of which are available in 2D and 3D. The cropped one only contains targets within the predefined tracking area used in their experiments, while the complete one contains annotations for all visible

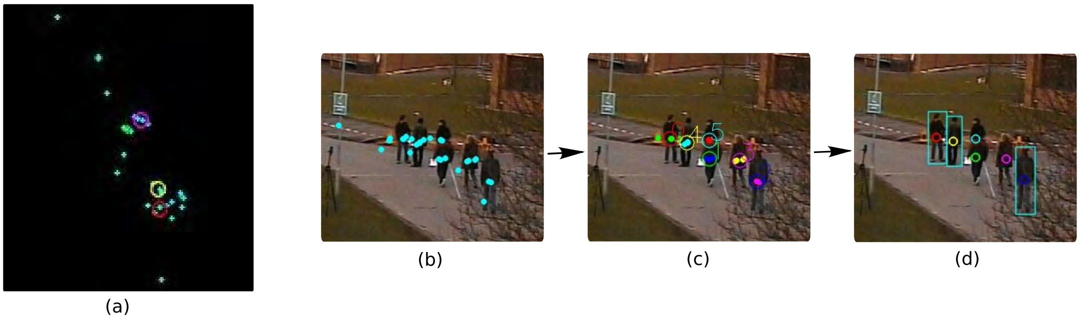


FIGURE 5. Fusion of reconstructions at one time step taking the PETS/S2.L1 [11] dataset as an example. (a): fusion seen from the top view: blue dots represent original reconstructions. Large circles group the fused reconstructions; (b-d): the corresponding fusion process shown on an image. (b): blue dots are original projections; (c): large circles group the fused projections and the color shows identities; (d): circles are fused detection centers, while the rectangles are original 2D detections by DPM on that image.

targets in the first view. We use the complete ground truth tracks and the same evaluation program provided by [30]. Even during the period of occlusion, the individual objects are labeled. Due to inaccurate localization and mistakes by the annotator, however, the bounding boxes are not always perfectly aligned. In the annotation, if a person leaves and enters the field of the first view, a new ID is assigned. Hence, there are in total 18 annotated persons in the ground truth. Note that since our 3D tracking approaches are based on 3D detections, the IDs are assumed to be retained as in the above mentioned case. This potentially causes higher numbers of ID switches of our trackers evaluated on this ground truth, for consistency of comparison, although smaller numbers would be more accurate. Moreover, we note that evaluation results of single-view based tracking using the 2D ground truth tracks in that view do not take into account calibration errors. In contrast, tracking in 3D space as in our case carries calibration errors from the multi-camera system. Hence the evaluated precision of the tracker, *i.e.* MOTP, would be worse than the real assessment. All in all, our evaluation gives a valuable comparison with other algorithms.

4) DETECTION

We obtained person detections in an image space for each camera video by *Deformable Part Models* [41]. We use the provided source code of DPM [42], where the classifier was only trained with people in general cases. Each of the acquired human detection is composed by a rectangle including the top left point, width and height, and a confident score.

B. EVALUATION

Since we use the same ground truth data as [30], comparison with their work is useful. In the following section, we present the evaluation of our approach compared with other methods. Since not many works reported tracking results based on multiple cameras, several state-of-the-art single-view based approaches are also taken into account. To have a fair comparison of the tracker, the performance of detections is evaluated as well. As a result, the difference of the evaluated scores between detection and tracking is used as an additional assessment metric. Afterwards, contributions of different combinations of cameras to the tracker are studied.

For the dataset of PETS2009 S2/L1, 1000 units in the world coordinate system on the ground plane equals 1m in the real world [30]. We assume that the mean distance between two persons is 0.5 meters. Thus the threshold for fusion is intuitively set as $\alpha = 500$. For all experiments, we set $w_{\text{comp}} = 1000$, $d\prime_{\text{penalty}} = 6000$, $d\prime\prime_{\text{penalty}} = d\prime_{\text{penalty}} - 120$. A constant number of 100 particles were used for each tracker in the second stage. If a tracker has not been updated for $\text{N} = 2$ frames, it will be terminated. Tracklets that are temporarily near not more than $\beta = 10$ could be linked as Equ. 14 shows.

Fig. 5 illustrates the fusion process from the top view and on one image frame. In Fig. 5(a), the fusion of groups are viewed from the top view. Small filled dots represent center positions of original reconstructions from all views. Large hollow circles group the ones belonging to identical objects. It shows that fusion recovers missing detections in certain views by using data from other cameras as well as removes false positive detections. In Fig. 5(b), blue filled dots represent projections $\mathbf{X}_t$ from original reconstructions of all the cameras. Afterwards, projections belonging to identical objects are fused into groups that are visualized by larger hollow circles shown in Fig. 5(c). Projections with large errors or false positives from certain views are un-grouped and hence are automatically rejected. In Fig. 5(d), rectangles show original 2D detections in the image and circles represent centers of grouped detections $\mathbf{X}'_t$, by which we can see that missing detections are recovered.

The quantitative results compared with the state-of-the-art algorithms are shown in Table 1. They indicate that our method outperforms others with the highest MOTA except the work of Hofmann *et al.* [14] that tuned optimal parameters from the ground truth data of the sequence. While MOTP is more related to the labeling precision of ground truth data, ours is comparable to the providers' in [30]. Furthermore, it could be observed that methods (final three rows of Table 1) involving separate improvement stages are more

TABLE 1. Quantitative results on PETS'09 S2.L1 sequence. We compared our approach with probabilistic tracking [43], particle filter based tracking-by-detection [6], bipartite matching [28], energy minimization [30], k-shortest paths [29], [44], two-stage graph [16], and joint optimization [12]. Only temporal distance is considered in weights function of *Our Approach*$^+$. Hofmann *et al.* [14]* tuned parameters from the ground truth.

Method	MOTP	MOTA	False Pos. Rate	Miss Rate	Id switches
Hofmann *et al.* [14]*	83.0%	99.4%	*n/a*	*n/a*	2
Yang *et al.* [43]	53.8%	75.9 %	*n/a*	*n/a*	*n/a*
Breitenstein *et al.* [6]	56.3%	79.7%	*n/a*	*n/a*	*n/a*
Bredereck *et al.* [28]	**85.4**%	74.9%	*n/a*	*n/a*	106
Andriyenko *et al.* [30]	76.1%	81.4%	*n/a*	*n/a*	15
Berclaz *et al.* [44]	56.0%	82.0 %	*n/a*	*n/a*	*n/a*
Berclaz *et al.* [29]	62.0 %	75.0 %	*n/a*	*n/a*	*n/a*
Jiang *et al.* [16]	81.44%	77.74%	7.83%	13.91%	24
Leal-Taix'e *et al.* [12]	60.0 %	76.0 %	*n/a*	*n/a*	*n/a*
Our Approach	76.21 %	83.08%	7.63%	8.39%	42
Our Approach$^+$	75.7%	**85.29**%	6.43%	7.44%	39

flexible and yield higher performance if the strategies in each step are well designed. To reveal the influence of different weighting functions in graph for tracklet linking, we also applied different weights while using the same tracklets for *Our Approach* and *Our Approach*$^+$. We set $\theta = 600$ and $\theta = 800$ for *Our Approach* and *Our Approach*$^+$, respectively. However, *Our Approach*$^+$ did not use spatial distances for the finite weights in Equ. 14 and obtained a better MOTA. This reveals that different distances used in the weighting function have distinct effectiveness. The reason of the higher MOTA *Our Approach*$^+$ obtained may be due to non-linear spatial motion of the objects confuses the algorithm in selecting the shortest paths of *Our Approach*. The work in [28] has a higher MOTP than ours, however, their MOTA is lower. In many applications, *e.g.* biomedical image analysis, MOTA attracts more attention since the consistency of object identities is more important. Above all, results show the advantages of our framework for multi-object tracking using multi-camera systems.

1) TRACKING VS. DETECTION PERFORMANCE

As indicated, tracking has a strong dependence on the detection performance in the tracking-by-detection framework. Thus assessment of tracking performance, with respect to the detection quality, provides an insight into how robust the tracker is with regard to detections.

Although there might be bias in the evaluation of tracking due to various reasons, *e.g.* distinct ground truth data, different detections, and varying reconstruction methods, the differences (D-values) between MODA and MOTA and between MODP and MOTP gives a quantitative assessment of tracker robustness. To some extent it reveals the improvement of the tracker given the detections. Fig. 6 shows a bar plot for the four mentioned metrics of [28], *Across-View* Followed by *Across-Time* Association, and [16] trackers compared with several the state-of-the-art approaches that are publicly available. For other methods, we would like to refer to [38]. In our experiments, only detections from DPM are used for tracking. The D-values of MOTP vs. MODP and MOTA vs. MODA can be seen as the differences of bars. From the figure we can see that most of the D-values of the other approaches are negative. D-values with respect to precision (MODP vs. MOTP) of our three approaches are positive, which means our trackers improve the localization precision of the objects. Our D-values regarding accuracy (MODA and MOTA) are negative as well, of which the absolute value is smaller than others, *e.g.* [6]. D-values validate the robustness of the presented approaches.

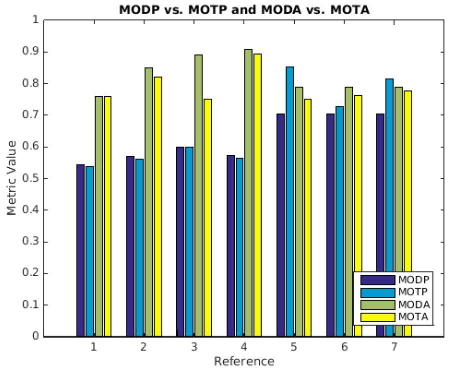


FIGURE 6. Detection performance vs. tracking performance of different approaches on the PETS/S2.L1 dataset [11]. Reference 1: [43]; Reference 2: [44]; Reference 3: [6]; Reference 4: [36]; Reference 5: [28]; Reference 6: our *Across-View* Followed by *Across-Time* Association approach; Reference 7: [16].

TABLE 2. Performance of the fused 3D detections tested on the PETS/S2.L1 database [11] with respect to α.

Parameter α	MODP	MODA	False Pos.	False Neg.
500	70.4%	65.2%	268	1350
600	70.5%	71.38%	198	1133
700	69.8%	73.59%	189	1039
800	70.0%	75.42%	174	969
900	70.4%	77.51%	150	896
1000	**70.6**%	78.58%	**147**	849
1100	70.4%	**78.75**%	157	**831**

C. ANALYSIS

As can be seen in Fig. 6, detection performance heavily affects the final tracking assessment. Since original reconstructions are fused first to obtain 3D detections used in the tracker, studies on the robustness of the fusion stage (Section III-B) is of significant importance. In the following, detection performances with respect to the parameter α and the number of cameras used are studied.

1) INFLUENCE OF PARAMETERS

The parameter α is the maximally allowed Euclidean distance between two reconstructions that might be clustered. It represents how much back-projection error can an identical object have, since there is calibration error for every camera. As shown in Table 2, MODA and MODP are affected by α in the PETS/S2.L1 dataset. Detections from all the cameras were used in the experiments. The larger α simplifies calculations for our algorithm in reconstructing camera detections in order to form single hypotheses. Smaller α makes accurate

TABLE 3. Performance of the fused 3D detections for the PETS/S2.L1 database [11] with respect to the cameras used.

Views	MODP	MODA	FP	FN
$3, 5, 7$	68.9%	31.48%	32	3154
$3, 5, 6$	68.0%	35.35%	46	2960
$3, 5, 6, 7$	66.8%	49.01%	62	2309
$5, 6, 7, 8$	60.4%	50.69%	93	2200
$3, 5, 6, 7, 8$	64.8%	58.37%	101	1835
$1, 3$	73.8%	27.63%	**8**	3357
$1, 5, 6$	72.5%	55.12%	27	2060
$1, 3, 5, 6$	71.9%	67.68%	55	1448
$1, 3, 7, 8$	70.3%	64.49%	96	1555
$1, 3, 5, 6, 7, 8$	70.4%	**78.75%**	157	831
1	**75.6%**	76.52%	430	**662**

reconstructed detections from multiple cameras difficult for our algorithm to cluster, which results in false negative detections. As can be seen from the table, if $\alpha = 1100$ the total performance is comparable to the performance of $\alpha = 1000$. The difference is that the number of false negatives of the former is smaller than the latter, while the number of false positives of the latter is better than the former. This is intuitive since more detections can be clustered into groups resulting in a reduced missing number as α increases. In this case, however, false positive reconstructions are also fused into groups. Notably, MODP remains nearly the same by changing the value of α.

2) INFLUENCE OF THE NUMBER OF CAMERAS

Since cameras are distributed in different locations in the scene and some of them have overlapped field of views while some may not, the contribution of each camera to the final tracking is different and varies at different time steps. Adopting different combinations of the cameras may generate distinct 3D detections. For this purpose, different numbers and combinations of the cameras are randomly selected to produce the clustered 3D detections for the PETS/S2.L1 dataset. Table 3 shows the evaluation of the detections. For all the experiments in the table, we set $\alpha = 1100$.

Three observations can be made from the table. First, the combinations of cameras containing the first view obtain higher MODPs. This can be explained by the fact that the ground truth tracks from the first view are used for evaluation, which reduces the influence of projection errors from other cameras. Second, as can be seen from the first five rows and from the following five rows, the usage of more cameras obtains higher detection performance. This performance follows logically. Third, when only the detections in the first view are used for reconstruction without fusing other views, MODP is the highest since no projection error from other views influences precision. The usage of the first view obtained a lower but comparable MODA in the case of using all of the cameras, but with a higher false positive number and a lower missing number of detections. In this case, the false positive reconstructions in the first view may not appear in one or more of the other views. Hence, the detection performance is not only related to the number of cameras used but also which cameras are used.

TABLE 4. Performance comparison on the PETS/S2.L1 database [11] with and without adding tracked positions from the 3D particle filter to the data fusion process.

Fusion	MODP	MODA	FP	FN
without tracked pos.	70.4%	78.75%	**157**	831
with tracked pos.	70.4%	**86.82%**	244	**369**

In addition for our *Across-View* Followed by *Across-Time* Association approach, we add the tracked positions from the 3D particle filter (Section III-C) to the fusion process. As shown in Table 4, detection performance increases by adding the tracked results as supplementary detections for fusion, since the number of false negatives is decreased to a large extent. We conclude that the increased number of false positives caused by adding tracked positions is likely related to the inaccuracy of tracked positions, or the false positive tracks themselves.

3) RUN TIME

The computation complexity of the proposed data fusion algorithm is nearly linear to the number of nodes in the graph regardless the number of cameras in the sensor network, which can achieve a real-time performance in practice. The multi-dimensional assignment for multi-sensor data fusion proposed in [13] is NP-hard when there are more than three cameras. Including the data fusion phase, our approach took approximately 1 *s*/f in the first 100 frames.

The computation complexity of the greedy matching in [28] is $\mathcal{O}(n^2 \log n)$, where n is the size of the nodes. The experiments were performed on one single core of an Intel Core2Quad$^{\text{TM}}$ CPU with 2.4 GHz and 8 GB of memory. On average, the running time for the first 50 frames of the first view is 2.98 *s*/f with and 2.52 *s*/f without using the online classifier. The runtime of the single-camera tracking increases to 8.2 *s*/f if the time of creating new tracks and training the online classifier is considered. Reference [28] has a running time of 38.0 *s*/f with and 36.6 *s*/f without utilization of the online classifier in the single-camera tracking. Note that they perform the system in individual cameras without any parallel computing and the calculation of the likelihood values does not use parallelism as well.

The runtime of [6] which is a single-camera tracking approach was 0.5 - 2.5 *s*/f without considering the time for detection calculation. The worst case complexity of [29] is $\mathcal{O}(k(m + nlogn))$, where k is the number of objects, m is the number of edges, and n is the number of nodes in the graph. Other works did not post their run time performance.

V. CONCLUSION

In this paper, we proposed a multi-object tracking framework to solve data fusion and data associations problems for unconstrained VSNs. Unlike sensor network topology inference-based trackers, which have limited flexibility, a novel graph-based data fusion modeling approach is conducted. We demonstrated a methodology to efficiently fuse data from separate sensors, and illustrated the process of enhancing the performance of 3D observations. The data fusion algorithm has a nearly linear computational complexity to the number of nodes in the graph regardless the number of cameras in the sensor network. Moreover, tracking multiple objects using particle filters makes it possible to incorporate motion information into object state estimation. Combining occlusion reasoning rules, tracklets generated are more reliable in difficult situations. Finally, global temporal and spatial features were used in the weighting function, to further link tracklets to form full tracks. The framework is generic and prior knowledge of specific scenarios is incorporated into the weights of the graphs. In addition, discussion of tracking performance with respect to different combinations of cameras gives insight to the coverage problem of VSNs.

The proposed approach is unsupervised, but learning optimal parameters from the dataset, *e.g.* velocity of the object or object appearance changes in different frames and cameras, will improve the results. It is not currently clear to which extent this learning process enhances our results. System robustness could be improved by another model different from occlusion reasoning.

REFERENCES

[1] P.-F. Wu, F. Xiao, C. Sha, H. Huang, R.-C. Wang, and N.-X. Xiong, "Node scheduling strategies for achieving full-view area coverage in camera sensor networks," *Sensors*, vol. 17, no. 6, p. 1303, 2017.

[2] J. Liu, S. Sridharan, and C. Fookes, "Recent advances in camera planning for large area surveillance: A comprehensive review," *ACM Comput. Surv.*, vol. 49, no. 1, p. 6, 2016.

[3] A. T. Kamal, J. H. Bappy, J. A. Farrell, and A. K. Roy-Chowdhury, "Distributed multi-target tracking and data association in vision networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 7, pp. 1397–1410, Jul. 2016.

[4] N. Xiong, R. W. Liu, M. Liang, D. Wu, Z. Liu, and H. Wu, "Effective alternating direction optimization methods for sparsity-constrained blind image deblurring," *Sensors*, vol. 17, no. 1, p. 174, 2017.

[5] H. Zannat, T. Akter, M. Tasnim, and A. Rahman, "The coverage problem in visual sensor networks: A target oriented approach," *J. Netw. Comput. Appl.*, vol. 75, pp. 1–15, Nov. 2016.

[6] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool, "Online multiperson tracking-by-detection from a single, uncalibrated camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1820–1833, Sep. 2011.

[7] C. Huang, B. Wu, and R. Nevatia, "Robust object tracking by hierarchical association of detection responses," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2008, pp. 788–801.

[8] Z. Wu, "Occlusion reasoning for multiple object visual tracking," Ph.D. dissertation, School Arts Sci., Boston Univ., Boston, MA, USA, 2013.

[9] R. Eshel and Y. Moses, "Homography based multiple camera detection and tracking of people in a dense crowd," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.

[10] R. Mohedano and N. I. A. Garc, "Robust multi-camera 3D tracking from mono-camera 2D tracking using Bayesian association," *IEEE Trans. Consum. Electron.*, vol. 56, no. 1, pp. 1–8, Feb. 2010.

[11] J. Ferryman and A. Shahrokni, "PETS2009: Dataset and challenge," in *Proc. IEEE Int. Workshop Perform. Eval. Tracking Surveill. (PETS-Winter)*, Dec. 2009, pp. 1–6.

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

[13] A. B. Poore, "Multidimensional assignment formulation of data association problems arising from multitarget and multisensor tracking," *Comput. Optim. Appl.*, vol. 3, no. 1, pp. 27–57, 1994.

[14] M. Hofmann, D. Wolf, and G. Rigoll, "Hypergraphs for joint multi-view reconstruction and multi-object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 3650–3657.

[15] J. Xing, H. Ai, and S. Lao, "Multi-object tracking through occlusions by local tracklets filtering and global tracklets association with detection responses," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 1200–1207.

[16] X. Jiang, M. Simon, Y. Yang, and J. Denzler, "Multi-marker tracking for large-scale X-ray stereo video data," *Signal Process., Image Commun.*, vol. 59, pp. 140–149, Nov. 2017.

[17] L. Zhang, Y. Li, and R. Nevatia, "Global data association for multi-object tracking using network flows," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.

[18] H. Aghajan and A. Cavallaro, *Multi-Camera Networks: Principles and Applications*. Orlando, FL, USA: Academic, 2009.

[19] A. A. Butt and R. T. Collins, "Multi-target tracking by Lagrangian relaxation to min-cost network flow," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Jun. 2013, pp. 1846–1853.

[20] J. Ma, J. Zhao, J. Tian, A. L. Yuille, and Z. Tu, "Robust point matching via vector field consensus," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1706–1721, Apr. 2014.

[21] S. M. Khan and M. A. Shah, "A multiview approach to tracking people in crowded scenes using a planar homography constraint," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2006, pp. 133–146.

[22] K. Kim and L. S. Davis, "Multi-camera tracking and segmentation of occluded people on ground plane using search-guided particle filtering," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2006, pp. 98–109.

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

[24] L. Wen, Z. Lei, M.-C. Chang, H. Qi, and S. Lyu, "Multi-camera multi-target tracking with space-time-view hyper-graph," *Int. J. Comput. Vis.*, vol. 122, no. 2, pp. 313–333, 2017.

[25] M. C. Liem and D. M. Gavrila, "Joint multi-person detection and tracking from overlapping cameras," *Comput. Vis. Image Understand.*, vol. 128, pp. 36–50, Nov. 2014.

[26] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Autom. Control*, vol. AC-24, no. 6, pp. 843–854, Dec. 1979.

[27] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE J. Ocean. Eng.*, vol. OE-8, no. 3, pp. 173–184, Jul. 1983.

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

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

[30] A. Andriyenko and K. Schindler, "Multi-target tracking by continuous energy minimization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 1265–1272.

[31] S. Wang and C. C. Fowlkes, "Learning optimal parameters for multi-target tracking with contextual interactions," *Int. J. Comput. Vis.*, vol. 122, no. 3, pp. 484–501, 2017.

[32] S.-H. Bae and K.-J. Yoon, "Confidence-based data association and discriminative deep appearance learning for robust online multi-object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 3, pp. 595–610, Mar. 2018.

[33] J. Bang-Jensen and G. Gutin, *Digraphs: Theory, Algorithms and Applications* (Springer Monographs in Mathematics). New York, NY, USA: Springer, 2009.

[34] M. Isard and A. Blake, "Condensation—Conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, 1998.

[35] X. Jiang, E. Rodner, and J. Denzler, "Multi-person tracking-by-detection based on calibrated multi-camera systems," in *Proc. Int. Conf. Comput. Vis. Graph.*, 2012, pp. 743–751.

[36] A. Andriyenko, K. Schindler, and S. Roth, "Discrete-continuous optimization for multi-target tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 1926–1933.

[37] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *J. Image Video Process.*, vol. 2008, Feb. 2018, Art. no. 1.

[38] A. Ellis, A. Shahrokni, and J. M. Ferryman, "PETS2009 and winter-PETS 2009 results: A combined evaluation," in *Proc. IEEE PETS Winter-PETS Workshop*, Dec. 2009, pp. 1–8.

[39] J. F. Henriques, R. Caseiro, and J. Batista, "Globally optimal solution to multi-object tracking with merged measurements," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 2470–2477.

[40] A. Andriyenko. (2015). *The Ground Truth of PETS/S2 Dataset*. [Online]. Available: http://research.milanton.de/data.html

[41] P. Felzenszwalb, R. B. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 2241–2248.

[42] P. Felzenszwalb. (2012). *Source Code of Deformable Part Models*. [Online]. Available: http://www.cs.berkeley.edu/~rbg/latent/

[43] J. Yang, Z. Shi, P. A. Vela, and J. Teizer, "Probabilistic multiple people tracking through complex situations," in *Proc. IEEE Workshop Perform. Eval. Tracking Surveill.*, Jun. 2009, pp. 79–86.

[44] J. Berclaz, F. Fleuret, and P. Fua, "Multiple object tracking using flow linear programming," in *Proc. IEEE Int. Workshop Perform. Eval. Tracking Surveill. (Winter-PETS)*, Dec. 2009, pp. 1–8.

XIAOYAN JIANG received the Ph.D. degree in computer science from Friedrich-Schiller-Universität Jena, Jena, Germany. She is currently a Lecturer with the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China. Her research interests include sensor data fusion, multi-object tracking for autonomous driving and visual surveillance, probability theory, and optimization algorithms. She has published numerous SCI/EI papers in the field of computer vision. She received the fund from National Natural Science Foundation of China in 2017. She received the scholarships from both Chinese government and German academic exchange service.

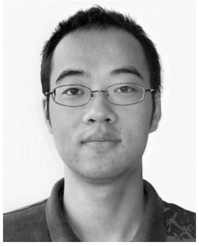
ZHIJUN FANG received the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China. He is currently a Professor and the Dean of the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science. His current research interests include image processing, video coding, and pattern recognition. He was a recipient of the GanPo 555 Talents Program, the One-Hundred, the One-Thousand, and the Ten-Thousand Talent Project award of Jiangxi province. He was a General Chair of Joint Conference on Harmonious Human Machine Environment 2013 and a General Co-Chair of International Symposium on Information Technology Convergence in 2014, 2015, 2016, and 2017.

NEAL N. XIONG received the both Ph.D. degrees from Wuhan University in sensor system engineering and from the Japan Advanced Institute of Science and Technology in dependable sensor networks. He is currently an Associate Professor (3rd year) with the Department of Mathematics and Computer Science, Northeastern State University, Tahlequah, OK, USA. Before he attended Northeastern State University, he was with Georgia State University, Wentworth Technology Institution, and Colorado Technical University (Full Professor about five years) about 10 years. His research interests include cloud computing, security and dependability, parallel and distributed computing, networks, and optimization theory. He has published over 200 international journal papers and over 100 international conference papers. Some of his works were published in the IEEE JSAC, the IEEE or ACM transactions, ACM Sigcomm workshop, the IEEE INFOCOM, ICDCS, and IPDPS. He was a recipient of the Best Paper Award in the 10th IEEE International Conference on High Performance Computing and Communications and the Best Student Paper Award in the 28th North American Fuzzy Information Processing Society Annual Conference. He has been a General Chair, a Program Chair, a Publicity Chair, PC member, and OC member of over 100 international conferences, and as a Reviewer of about 100 international journals, including the IEEE JSAC, the IEEE SMC (Park: A/B/C), the IEEE Transactions on Communications, the IEEE Transactions on Mobile Computing, the IEEE Transactions on Parallel and Distributed Systems. He is serving as an Editor-in-Chief, an Associate editor, or an Editor member for over 10 international journals (including Associate Editor for the IEEE Transactions on Systems, Man and Cybernetics: Systems, an Associate Editor for *Information Science*, an Editor-in-Chief for the *Journal of Internet Technology*, and an Editor-in-Chief for the *Journal of Parallel & Cloud Computing*), and a Guest Editor for over 10 international journals, including *Sensor Journal*, WINET, and MONET.

YONGBIN GAO received the Ph.D. degree from Chonbuk National University, South Korea. He is currently a Faculty Member with the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China. He has published numerous SCI papers in prestigious journals, such as *Information Science*, Pattern Recognition Letters, in the area of image processing, pattern recognition, and computer vision.

BO HUANG was born in China, in 1985. He received the M.S. and Ph.D. degrees in computer science from Wuhan University in China. He is currently a Lecturer with the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China. His research interests include artificial intelligence, software engineering, and formalization method.

JUAN ZHANG received the B.S. and M.S. degrees from Jiangxi Normal University, China, in 1997 and 2005, respectively, and the Ph.D. degree from Shanghai University, China, in 2012, all in computer science. From 2012 to 2014, she was a Post-Doctoral Fellow with the School of Communication and Information Engineering, Shanghai University. She is currently an Associate Professor with the College of Electronic and Electrical Engineering, Shanghai University of Engineering Science, China. Her research interests include software testing and computer graphics.



LEI YU received the B.E. degree from East China Normal University, Shanghai, China, in 2007, and the Ph.D. degree from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2015. He is currently a Lecturer with the Shanghai University of Engineering Science, Shanghai, China. His research interests include multimedia content analysis, computer vision, and machine learning.

PATRICK HARRINGTON is currently pursuing the Ph.D. degree in computer science from Oklahoma State University. He is currently a tenured Associate Professor with Northeastern State University, Tahlequah, OK, USA. He is currently involved in research in the area of artificial intelligence and networks and enjoys working with students in undergraduate research.