Multi-view Multi-object Tracking Based on Global Graph Matching Structure

Chao Li, Shantao Ping, Hao Sheng, Jiahui Chen, and Zhang Xiong

State Key Laboratory of Software Development Environment, School of Computer Science and Engineering Beihang University, Beijing 100191, P. R. China
{pingst,licc,shenghao,chenjh}@buaa.edu.cn

Abstract. We present a novel global graph matching framework based on virtual nodes for multi-object tracking in multiple views. Contrary to recent approaches, we incorporate a global graph matching structure (GGMS), allowing the tracker to better cope with long-term occlusions and tracking failure caused by interaction of targets. In our approach, the matching problem is solved as follows: Virtual detections are introduced by mapping the nodes among views, to ensure that the amount of detections in each view is the same, and then realize the whole graph matching. In addition, appropriate optimization is performed to convert this mapping problem to the Assignment Problem, which could be efficiently addressed by the Hungarian Algorithm. Finally, we demonstrate the validity of our approach on the publicly available datasets, and achieve very competitive results by quantitative evaluation.

Keywords: Multi-object tracking, multi-view, graph matching

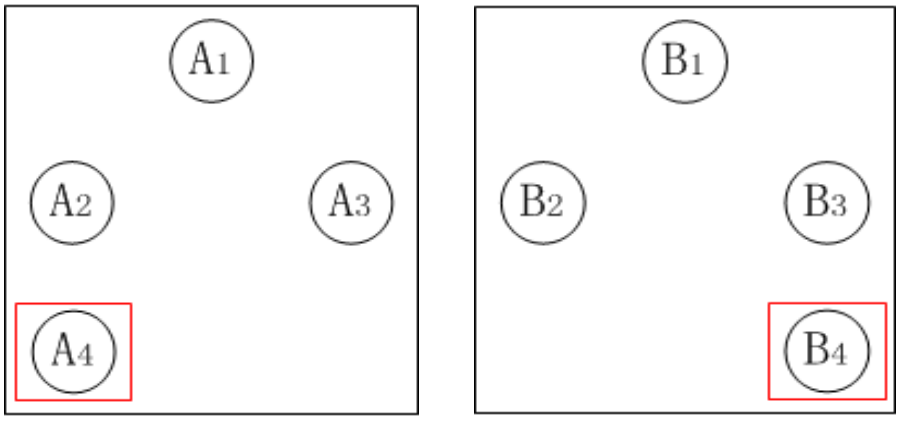
1 Introduction

With the fast development of smart devices, numerous cameras lead to ubiquitous video sources. Crowd-sourced video retrieval systems [1] based on video content comparison has emerged. Multi-object tracking is a key problem for many computer vision tasks, such as surveillance [2], animation or activity recognition. The tracking in video consists of detecting all subjects in every frame, and following their complete trajectory over time. Successful research on a new generation of reliable pedestrian detectors [3, 4] has prompted the use of the tracking-by-detection paradigm [5], even for crowded or semi-crowded scenarios. Under this paradigm, the problem is often divided in two steps: detection and data association [6, 7]. The tracker first acquires a set of detections using a pedestrian detector. The individual detections are then assigned to tracks, where each track is composed of all the detections from a single individual. If all persons were to be correctly observed at every timestamp this task would be trivial, however, due to false positive detections, occlusions and missed detections, this association problem becomes very challenging.

Usually, the problem of tracking is divided into two directions: monocular tracking and multi-view tracking. In recent research, the minimum-cost network flow tracking

approach [8, 9] is more popular in monocular tracking. This method can effectively cope well with short-term occlusion, however, it tends to become unreliable when the long-term occlusion occurs.

Unlike monocular tracking, in multiple views, the information from other perspective can complement better the detection errors in the main view, which may be caused by occlusions or detection failures. While considering multi-view tracking, the additional problem of data association between views arises. Reconstruction and tracking are two main problems. Wu et al. [10] handled these works as separate stages. Leal et al. [11] attempted to jointly solve these two problems for multi-view multi-object tracking. Although excellent results have been achieved in this method, there still exists potentiality in dealing with the occlusion and missing detections.



(a) view1 (b) view2

Fig. 1. These two graphs represent the 3D detections in view 1 and view 2 corresponding to the F frame. We assume that A_1, A_2, A_3 and B_1, B_2, B_3 are matched well with each other respectively, and A_4 can`t match with B_4.

To achieve this promotion, we continue the work of [11] which has discussed above. In our approach, we propose a method to solve the problem iteratively. Firstly, the output of [11], such as the world coordination of detections, is our input. Virtual nodes which are obtained by mapping each of the detections in each view to another are introduced to ensure that the amount of detections in each of the two views is same. Then a weight defined as the distance between different nodes and some constraints for the graph are introduced, in order to further realize the matching of the two graphs.

The rest of the paper is organized as follows. Section 2 presents related work. The formulation of our proposed method is described in section 3. Section 4 describes an optimization approach to solve the problem. Next, experiments are presented in Section 5 and, finally, the paper is concluded in Section 6.

2 Related Work

Object tracking has been studied extensively. For example, Kang et al. [12] proposed to take the multi-camera tracking as a problem of maximum joint probability model based on color. By estimating the object model through Kalman filtering, it used the joint probabilistic data filtering and multi-camera homography to multi-target tracking. Lien et al. [13] presented a tracking method for multi-view object based on the cooperation of hidden Markov process and particle filtering. Nummiaro et al. [14] put forword a tracking algorithm for multi-view object based on particle filtering, but unlike the idea of information fusion, this algorithm selected the best point of view for object tracking among the different perspectives.

Many global approaches that use more information have been explored to overcome occlusion and detector failure. Leal et al. [11] attempted to jointly solve these two problems for multi-view multi-object tracking. In this work, a separate tracking graph is constructed for each view. In addition, for each pair of views, an additional tracking graph is constructed, providing the coupling constraints for the involved views. In [15], it presents a solution which only requires a single tracking graph. Multi view coupling constraints are incorporated into the reconstruction nodes within the tracking graph. Tracking therefore only needs to be done once in the world coordinate space.

3 The Formulation of the Global Graph Matching Structure(GGMS)

In this section, the problem in [11] and the methods proposed in this paper is described.

As stated earlier, we continue the work of [11] which has proven to be a mathematically reliable framework for multi-object tracking. In [11], one tracking graph(2D layer) is constructed for each view and the multi-camera couplings(3D layer) are incorporated by an additional tracking graph for each possible camera pair in the world coordinate space. Ideally, an object which is seen by all available cameras generates a 2D detection in each view and the corresponding projections to the common world coordinates should all come to the same location. However, due to projection errors and imprecise detections, the resulting 3D positions are unlikely to match up exactly. In 3D layer, three types of edges, reconstruction edges, camera coherency edges and temporal 3D edges are introduced. Instead, these edges act as prizes for the graph, when the reconstruction, camera coherence and temporal 3D edges are sufficiently negative, it assigns the same identity to the objects seen by all available cameras. The problem lies in that when occlusion occurs, the 3D reconstruction will be based on only one visual angle. At this time, the occlusion could not be effectively resolved.

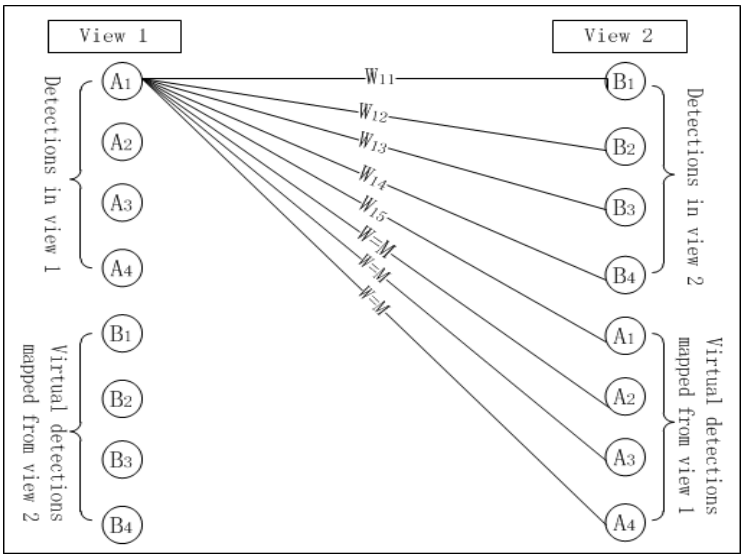


Fig. 2. This illustration shows how the detections and the virtual detections between two views matched with each other. The matching cost is expressed as W_{ij}, where W_{ij}= M means these two detections can`t be matched. And M is an infinite constant defined.

In this paper, we introduce a graph structure, and then solve the occlusion problem demonstrated above by the matching of the graph. As shown in Fig 1, each graph has four 3D detections corresponding to the F frame. Each 3D detection is defined by a tuple Ai = (id_i, x_i, y_i z_i), where id_i and (x_i, y_i, z_i) are the object identity and the location in world coordinates. It is assumed that detections A_1, A_2, A_3 shown in Fig 1(a) are matched with detections B_1, B_2, B_3 shown in Fig 1(b) respectively. The detection A_4 from view 1 cannot find corresponding matching point in view 2, and so is the detection B_4 from view 2. Our purpose is to find an algorithm to achieve the matching of the two graphs. Due to the existence of special circumstances: the amount of the detections in two views may be not same, so virtual nodes are introduced to ensure that the number of the detections is the same in each view as shown in Fig 2. W represents the weight of detections respectively, which can be expressed as

$$W_{ij} = e^{dist-\partial} - 1 \tag{1}$$

$$\text{dist} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \tag{2}$$

where dist is the Euclidean distance between detections in world coordinates and δ reflects the reasonable threshold which represents the maximum distance between two detections when they are correctly matched. We introduce an exponential function of the (dist − δ) which guarantees the W_{ij} is negative when matching and positive when not matching. The matching of two graphs indicates that the graph has the smallest weight.

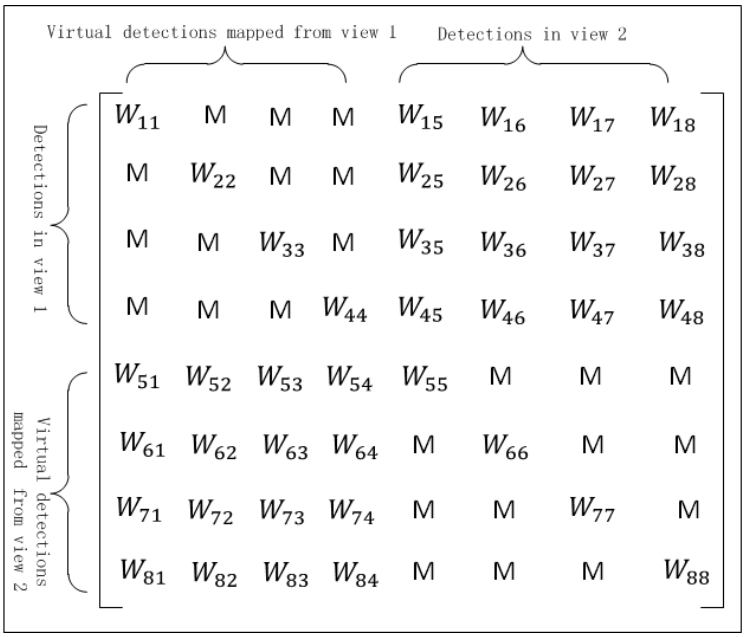


Fig. 3. This illustration shows that how the detections and the virtual detections between two views are matched with each other. The matching cost is expressed as W_{ij}, where W_{ij} = M means these two detections can`t be matched. And M is an infinite constant defined.

To get the smallest weight, we can convert it into a Linear Program (LP), its objective function is linearized with a set flags X_{ij} = {0,1} which indicate if an edge i → j is in the solution or not. We define a number of variables in advance, as follows: (i) Detections in view 1 are represented by i, among which, No. 1 to No. m are inherent, while the rest are virtual detections transformed from view 2. (ii)Detections in view 2 are represented by j, among which, No. 1 to No. m are virtual nodes mapped from view 1, while the rest are inherent. The proposed graph-matching structure can be expressed as a LP with the following objective function:

$$\mathrm{F} = \min \sum_{i=1}^{count} \sum_{j=1}^{count} X_{ij} W_{ij} \tag{3}$$

where count = m + n is the total amount of detections in view 1 and view 2. M is defined to represent the quantity of detection in view 1 and so as n in view 2. The problem is subject to the following constraints:

$$\sum_{j=m+1}^{count} X_{ij} = 1\,, \quad i = 1,2,\dots,m \tag{4}$$

The constraint ensures that each detection in view 1 has one (also the only one) corresponding detection in view 2 whether it is a detection or a virtual node mapped from view 1.

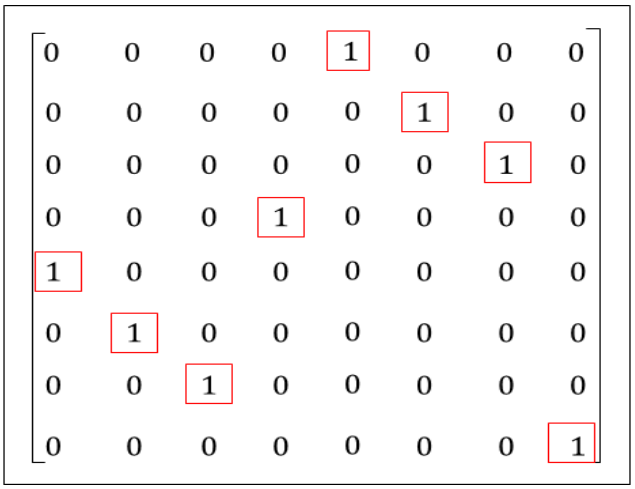


Fig. 4. This picture is a solution matrix, which is corresponded to the front of the weight matrix. In this matrix, element 1 means X_{ij}= 1. In other words, the two detections i and j are matched with each other. And all the elements of X_{ij}= 1 are in the final solution.

$$\sum_{i=1}^{m} X_{ij} = 1\,,\ \ j = m+1, m+2, \ldots, count \tag{5}$$

The constraint ensures that each detection in view 2 has one (also the only one) corresponding detection in view 1 whether it is a detection or a virtual node mapped from view 2.

$$\sum_{i=1}^{m} X_{ij} \leq 1\,,\ \ j = 1,2, \ldots, m \tag{6}$$

This guarantees that each virtual detection (mapped from view 2) in view 1 has at most one matching detection in view 2.

$$\sum_{j=m+1}^{count} X_{ij} \leq 1\,,\ \ i = m+1, m+2, \ldots, count \tag{7}$$

This assures that each virtual detection (mapped from view 1) in view 2 has at most one matching detection in view 1.

$$\sum_{i=1}^{m} X_{ij} = 1\,,\ \ j = 1,2, \ldots, m \tag{8}$$

$$\sum_{j=m+1}^{count} X_{ij} = 1\,,\ \ i = m+1, m+2, \ldots, count \tag{9}$$

In order to solve the Linear Programming problem more conveniently, we convert the constraint conditions of the inequalities (6) and (7) depicted above to the equalities (8) and (9) expressed below.

4 Optimization Approach

In this section, we mainly talk about how to make further optimization of the above problem, as well as how to solve this problem. Firstly, it is converted into the Assignment Problem. Then we solve it efficiently with the Hungarian Algorithm.

4.1 How to Convert the Problem

From the notations in section 3, the constraints of (4),(5),(8) and (9) can be optimized as the following equalities which can be expressed as:

$$\sum_{j=1}^{count} W_{ij} = 1\,,\ \ i = 1,2,\dots,count \tag{10}$$

$$\sum_{i=1}^{count} X_{ij} = 1\,,\ \ j = 1,2,\dots,count \tag{11}$$

where (10) denotes each detection (including the virtual detections mapped from view 2) in view 1 has one corresponding detection in view 2. And (11) represents each detection (contain the virtual detections mapped from view 1) in view 2 can only be arranged to one detection from view 1. The X_{ij} is re-expressed as below:

$$X_{ij} = 0\ or\ 1\,,\ \ i,j = 1,2,\dots,count, \tag{12}$$

The equalities (10), (11) and (12) constitute the new and optimal constraint of the problem. Then the objective function (3) with the optimal constraints has become as an Assignment Problem. The solution will be demonstrated in the next subsection.

4.2 The Solution Strategy

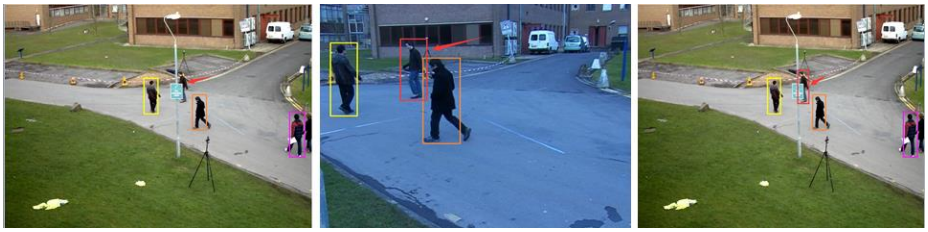
In order to solve this problem, we introduce a weight matrix, as shown in Fig 4. The row vector of the matrix represents the detection in view 1, where the former m line represents the point view 1, and the remainder denotes the virtual point mapped from view 2. And meanwhile, the column vector of the matrix represents the detection in view 2, where the former m column represents the virtual point mapped from view 1, and the remainder denotes the point in view 2.

Because the detection in the same view could not be matched with each other, we define X_{ij} = M (i, j = 1, 2, .., m and i, j = m + 1, m + 2, .., count and i ≠ j), which represents an infinite constant and is not likely to be contained in the solution. The weight matrix could be solved with the Hungarian Algorithm, and then a solution matrix (as in Fig 5) consists of 0 and 1 is obtained. In the solution matrix, 0 represents unselected edge of X_{ij} and 1 represents the matching edge.

Since we evaluate on view 1, and the second view we use does not show all the pedestrians. Therefore, we consider the detections of view 1 as the main detections and only use the second view to further improve the 3D position. In other words, if detections in view 2 are matching with the virtual points mapped from view 2, then we add the detections (corresponding to X_{ii} = 1, i = m + 1, m + 2, .., count in the solution matrix) to the final results of view 1.

5 Experiment

In this section, we show the tracking results of the proposed method on the key problem in computer vision, namely occlusion usually exists in multi-object tracking. Evaluating results of multi-object tracking is non-trivial because errors might be present in various forms including ID switches, broken tracks, imprecisely localized tracks and false tracks. Measures such as MOTA [16, 17] combine different errors into a single score and enable the global ranking of tracking methods.



(a) view 1 (b) view 2 (c) mapped result

Fig. 5. Tracking results on PETS 2009 for two cameras. (a) result of single view 1--- a pedestrian is occluded , (b) result of single view 2 (corresponding to view 6 in PETS 2009 S2L1) , (c) our result by mapping detections between view 1 and view 2 to solve the occlusion existed in view 1.

We show the tracking results (shown in Fig.5) of our approaches on the publicly available PETS2009 dataset [18], a scene with several interacting targets, based on MOTA metrics depicted above. We compare our results to some other multi-camera tracking methods [19, 20]. As shown in Table 1, our method generally has comparable MOTA, MOTP and recall scores with [11]. The result indicates that the occlusion could be addressed in this method more effectively and our objective function is easy to construct and solve.

Table 1. Table summarizing results over PETS 2009 S2.L1 sequence. Abbreviations are as follows GT - ground truth tracks. MT - Mostly tracked. PT - partially tracked. ML - mostly lost. Comparison of several methods tracking on a variable number of cameras.

Method	Camera Numbers	GT	MT	PT	ML	MOTA	MOTP	Prcn	Rcll
Berclaz[19]	5	--	--	--	--	75	62	--	--
Berclaz[20]	5	--	--	--	--	82	56	--	--
Leal[10]	3	--	--	--	--	71.4	53.4	--	--
Leal[10]	2	--	--	--	--	76.0	60.0	--	--
Ours	2	**19**	**19**	0	0	**93.2**	79.9	93.4	96.8

6 Conclusion

In this paper, we present a novel method for multi-object tracking in overlapping views. This new approach achieves the matching of two graphs, by introducing virtual nodes mapped from other corresponding view. An optimal solution is provided based on the Hungarian Algorithm. The result shows that this method can be used to complete large tracking gaps caused by occlusion or detector failure. The proposed method has shown good performance on the publicly available PETS 2009 S2.L1 sequence, solving the problem of occlusion and matching state of the art performance.

Acknowledgement

This study was partially supported by the National Natural Science Foundation of China (No. 61370122) and the National High Technology Research and Development Program of China (No. 2013AA01A603). Supported by the Programme of Introducing Talents of Discipline to Universities and the Open Fund of the State Key Laboratory of Software Development Environment under grant #SKLSDE-2015ZX-21. Thank you for the support from HAWKEYE Group.

References

1. Cihang Liu, Lan Zhang, Kebin Liu, and Yunhao Liu, “Scan without a glance: Towards content-free crowdsourced mobile video retrieval system,” in International Conference on Parallel Processing, 2015, pp. 250–259.
2. Alper Yilmaz, Omar Javed, and Mubarak Shah, “Object tracking: A survey,” Acm Computing Surveys, vol. 38,no. 4, pp. 81C93, 2006.
3. L. Bourdev and J. Malik, “Poselets: Body part detectors trained using 3d human pose annotations.,” in Proceedings / IEEE International Conference on Computer Vision. IEEE International Conference on Computer Vision, 2009, pp. 1365–1372.
4. Pedro F Felzenszwalb, Ross B Girshick, Mc Allester David, and Ramanan Deva, “Object detection with discriminatively trained part-based models,” Pattern Analysis Machine Intelligence IEEE Transactions on, vol.32, no. 9, pp. 1627–1645, 2010.
5. M.D. Breitenstein, F. Reichlin, B. Leibe, and E. KollerMeier, “Robust tracking-by-detection using a detector confidence particle filter,” in IEEE International Conference on Computer Vision, 2009, pp. 1515–1522.
6. Li Zhang, Yuan Li, and Ramakant Nevatia, “Global data association for multi-object tracking using network flows,” in Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, 2008, pp. 1–8.
7. Aleksandr V. Segal and Ian Reid, “Latent data association: Bayesian model selection for multi-target tracking,” in Computer Vision (ICCV), 2013 IEEE International Conference on, 2013, pp. 2904–2911.
8. Asad A. Butt and Robert T. Collins, “Multi-target tracking by lagrangian relaxation to min-cost network flow,” in IEEE Conference on Computer Vision Pattern Recognition, 2013, pp. 1846–1853.

9. Milan Anton, Roth Stefan, and Schindler Konrad, “Continuous energy minimization for multitarget tracking,” Pattern Analysis Machine Intelligence IEEE Transactions on, vol. 36, no. 1, pp. 58–72, 2014.
10. Zheng Wu, Nickolay I. Hristov, T. H. Kunz, and M. Betke, “Tracking-reconstruction or reconstructiontracking? comparison of two multiple hypothesis tracking approaches to interpret 3d object motion from several camera views,” in Motion and Video Computing, 2009. WMVC ’09. Workshop on, 2009, pp. 1–8.
11. Laura Leal-Taixe, Gerard Pons-Moll, and Bodo Rosen- hahn, “Branch-and-price global optimization for multiview multi-target tracking,” in Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, 2012, pp. 1987–1994.
12. J. Kang, I. Cohen, and G. Medioni, “Persistent objects tracking across multiple non overlapping cameras,” Proceedings of IEEE Workshop on Motion and Video Coputing, 2005, pp.112-119.
13. K. C. Lien, and C. L. Huang, “Multiview-based cooperative tracking of multiple human objects,” Journal of Image and Video Process, vol.8(2), pp.1-13, 2008.
14. K. Nummiaro, et al. “Color-based object tracking in multi-camera enviroment,” Proceedings of Pattern Recognition Symposium, 2003, pp.591-599.
15. Martin Hofmann, Daniel Wolf, and Gerhard Rigoll,“Hypergraphs for joint multi-view reconstruction and multi-object tracking,” in IEEE Conference on Computer Vision Pattern Recognition, 2013, pp. 3650–3657.
16. Anton Milan, Konrad Schindler, and Stefan Roth,“Detection- and trajectory-level exclusion in multiple object tracking,” in IEEE Conference on Computer Vision Pattern Recognition, 2013, pp. 3682–3689.
17. Martin Hofmann, Michael Haag, and Gerhard Rigoll, “Unified hierarchical multi-object tracking using global data association,” in Performance Evaluation of Tracking and Surveillance (PETS), 2013 IEEE International Workshop on. IEEE, 2013, pp. 22–28.
18. J. Ferryman and A. Shahrokni, “Pets2009: Dataset and challenge,” in Performance Evaluation of Tracking and Surveillance (PETS-Winter), 2009 Twelfth IEEE International Workshop on, 2009, pp. 1–6.
19. J. Berclaz, F. Fleuret, and P. Fua, “Multiple object tracking using flow linear programming,” in Performance Evaluation of Tracking and Surveillance (PETSWinter), 2009 Twelfth IEEE International Workshop on, 2009, pp. 1–8.
20. Jerome Berclaz, Francois Fleuret, Engin Turetken, and Pascal Fua, “Multiple object tracking using k-shortest paths optimization,” IEEE Transactions on Software Engineering, vol. 33, no. 9, pp. 1806–1819, 2011.