Solving Multiple People Tracking In A Minimum Cost Arborescence

Roberto Henschel
Institut für Informationsverarbeitung
Universität Hannover
henschel@tnt.uni-hannover.de

Laura Leal-Taixé
Institute of Geodesy and Photogrammetry
ETH Zürich
leal@geod.baug.ethz.ch

Bodo Rosenhahn
Institut für Informationsverarbeitung
Universität Hannover
rosenhahn@tnt.uni-hannover.de

1. Introduction

For many applications of computer vision, it is necessary to localize and track humans that appear in a video sequence. Multiple people tracking has thus evolved as an ongoing research topic in the computer vision domain.

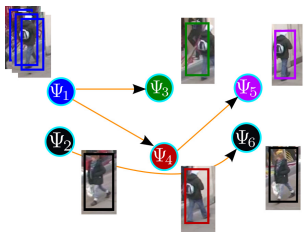
A commonly used approach to solve the data association problem within the tracking task is to apply a hierarchical tracklet framework [5]. Although there has been great progress in such a model, mainly due to its good bootstrapping capabilities, so far little attention has been drawn to improve the quality of the tracklets itself.

A main issue of the hierarchical frameworks, as used in the common literature, is that they make hard decisions at each iteration of the association step. Especially in ambiguous situations, tracklets are still being merged or removed so that the system is prone to error propagation. To avoid these problems, we propose a new framework where unreliable decisions are prevented. Instead, unclear aggregations are being postponed to a later iteration, when more information is available. To maintain the possible associations of tracklets in difficult situations, we propose a new trajectory model, which we call *tree tracklets* (see Fig. 1).

While recent multiple people trackers model the association problem mainly in a flow network (e.g. [6] [5]), we employ a rooted, directed and weighted graph $G = (V, E, w)$ which is of a simpler structure, in particular has fewer nodes and edges. Thereby, we obtain the global-optimal solution of each iteration in linear time in the number of nodes by computing a minimum cost arborescence.

2. An improved hierarchical tracklet model

In order to obtain reliable tracklets, we process the sequence several times, until we observe convergence or some threshold is reached. The node set $\underline{V}$ of the association



(a) Tree tracklets

Figure 1. A typical tree tracklet, maintaining possible aggregations. As the tracklets grow, they will contain enough information to solve ambiguous situations.

graph G consists of all detections in the first iteration and of all already computed tracklets in the ith iteration ($i > 1$), respectively. Finally, we add a virtual start node Λ, hence $V = \underline{V} \cup \{\Lambda\}$ is the node set of G.

To model the start of a trajectory, we set $\overline{E} := \{(\Lambda, \underline{v}) \mid \underline{v} \in \underline{V}\}$. All possible aggregations of tracklets are modeled in the edge set $\underline{E}$, that is, $\underline{E}$ connects each node $\underline{v} \in \underline{V}$ to those nodes $\underline{v}' \in \underline{V}$, where the first detection of the tracklet $\underline{v}'$ and the last detection of the tracklet $\underline{v}$ are within a predefined time-window T. Hence $E := \underline{E} \cup \overline{E}$ is the edge set of G.

The weights $w(u, v)$ between any two nodes are defined according to the affinities of the corresponding tracklets, where $w(\Lambda, v)$ for $v \in \underline{V}$ is defined as the likelihood of v being the start of a person's trajectory.

Now a *tree tracklet* Ψ is a connected subgraph $\Psi \subset G$ with indegree ≤ 1 in Ψ, for all of its nodes. Since there are no restrictions on the outdegrees of the nodes of Ψ, a tree tracklet can maintain several possible tracklet aggregations.

The purpose of the introduction of tree tracklets is to explain all current tracklets (or detections) as good as possible by connecting them in the most reasonable way. Thus

we use a maximum a posteriori probability estimate in order to get the best coverage of the association graph by tree tracklets, where we model the prior probability as a Markov chain, similar to [8]. In particular, we consider the negative log likelihoods of the affinities, so that a minimization returns the desired cover. It follows that the optimal tree tracklets coverage is obtained by computing the minimum cost arborescence $G' = (V, E', w)$ of G, and removing the edge set $\overline{E}$ from E' afterwards. Note that a minimum cost arborescence (MCA) can be seen as the minimum spanning tree problem, adapted to the case of a directed graph. Edmond's algorithm [2] provides a solution for rooted, directed and weighted graphs; the result is computed in $\mathcal{O}(n^2)$ in the number of nodes n of G. However, since the graph G is acyclic, the MCA G' can be obtained in $\mathcal{O}(n)$ (see [4]).

Once G' has been computed, we go through all nodes of G'. For $v \in V$, let $N_{\mathrm{f}}(v)$ denote the set of neighboring nodes of v, that appear at a later time, so $N_f(v) := \{v' \in V \mid (v, v') \in E'\}$. Now if the tracklet v is assigned to more than one tracklet, then we cut off all these links, since there is still an ambiguity which one is the correct aggregation. Hence we set $E' := E' \setminus \{(v, v') \mid v' \in N_f(v)\}$. The remaining tracklets are then the input for the next iteration or the final result, respectively.

For more details on this tracking framework, we refer the reader to [4].

3. Experiments

To evaluate the performance of the proposed framework, we compute the trajectories on several publicly available datasets: Bahnhof, Sunnyday [3] and TownCenter [1]. Details on the used detections are given in [4].

3.1. Tracking Performance

Since we implemented the tracker using 3D detections, we compare only to other state-of-the-art tracking frameworks that are based on such detections.

Table 1 shows that the performance of the proposed method is competitive with state-of-the-art tracking approaches. Especially on the semi-crowed TownCenter dataset, our method shows a 6% higher Recall and a 13% higher MT value, compared to the best method. At the same time, the number of identity switches is decreased. Note that our algorithm neither considers any special social modeling as in [7, 6] nor any type of appearance model.

3.2. Runtime

Our tracking system is implemented as a non-optimized Matlab code. On a sequence of 999 frames with 6536 detections and 5 iterations, trajectories are computed in 8 seconds on a 3.5 GHz machine (see Table 2). Having linear time complexity at each iteration, the computational complexity is $\mathcal{O}(Kd)$, when performing K iterations with d initial detections. Note however, that the number of tracklets decreases in most cases exponentially, resulting in a much better runtime complexity.

Dataset	Method	Rec.	Prec.	MT	PT	ML	Frg	Ids
Bahnhof	[8]	67.6	70.9	43.7	46.8	9.6	176	**39**
	[6]	73.3	75.4	51.1	41.5	7.4	**155**	107
	[7]	71.6	**84.9**	46.8	48.9	**4.3**	173	62
	Proposed	**75.3**	78.4	**56.4**	**38.3**	5.3	166	76
Sunnyday	[8]	76.8	70.6	**73.3**	**16.7**	10.0	39	**9**
	[6]	78.1	75.3	**73.3**	**16.7**	10.0	**29**	24
	[7]	75.5	**80.5**	66.6	23.3	10.1	33	15
	Proposed	**79.4**	75.9	**73.3**	20.0	**6.7**	38	12
TownCenter	[8]	77.2	79.9	53.3	39.6	**7.1**	135	233
	[6]	77.9	**88.7**	53.3	37.8	8.9	56	68
	[7]	74.5	77.5	44.0	48.4	7.6	**53**	42
	Proposed	**83.8**	81.2	**66.7**	**23.1**	10.2	70	**41**

Table 1. Tracking results on different datasets. MT = mostly tracked. PT = partially tracked. ML = mostly lost. Frg = fragmented tracks. Ids = identity switches. Code for [8, 7, 6] were provided by the authors of [6].

Method	Total runtime	Graph creation	Solver
Simplex [8]	16	9	7
Proposed	**8**	**6**	**2**

Table 2. Runtime in seconds of the algorithm on the Bahnhof dataset (999 frames).

References

[1] B. Benfold and I. Reid. Stable multi-target tracking in real-time surveillance video. In *CVPR*, 2011.

[2] J. Edmonds. Optimum branchings. *Journal of Research of the National Bureau of Standards Section B - Mathematical Sciences*, (4), 1967.

[3] A. Ess, B. Leibe, K. Schindler, and L. van Gool. A mobile vision system for robust multi-person tracking. In *CVPR*, 2008.

[4] R. Henschel, L. Leal-Taixé, and B. Rosenhahn. Efficient multiple people tracking using minimum cost arborescences. In *GCPR*, 2014.

[5] C. Huang, B. Wu, and R. Nevatia. Robust object tracking by hierarchical association of detection responses. In *ECCV*, 2008.

[6] L. Leal-Taixé, G. Pons-Moll, and B. Rosenhahn. Everybody needs somebody: Modeling social and grouping behavior on a linear programming multiple people tracker. *ICCV Workshops. 1st Workshop on Modeling, Simulation and Visual Analysis of Large Crowds*, 2011.

[7] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool. You'll never walk alone: modeling social behavior for multi-target tracking. In *ICCV*, 2009.

[8] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008.