Tracking Interacting Objects Using Intertwined Flows

Hongzhi Liu

July 4, 2018

Abstract

Tracking people or objects over time can be achieved by first running detectors that compute probabilities of presence in individual images and then linking high probability detections into complete trajectories. Most of recent approaches focus on one kind of object or only model simple interactions. Today, I read a thesis written by Xinchao Wang who is an IEEE Fellow. His team introduce a method that tracking different kinds of interacting objects can be formulated as a network-flow Mixed Integer. This is made possible by tracking all objects simultaneously using intertwined flow variables and expressing the fact that one object can appear or disappear at locations where another is in terms of linear flow constraints. Their proposed method is able to track invisible objects whose only evidence is the presence of other objects that contain them.

1. Overview of Tracking Model

Early approaches of the category focused on tracking a single object and relied on gating and Kalman filtering [6]. Because of their recursive nature, they are prone to errors such as drift which are difficult to recover from. Many approaches in this category aim at improving tracking accuracy by updating a classifier frame by frame. These techniques have proved effective for single object tracking. In recent years, techniques that optimize a global objective function over many frames have emerged as powerful alternatives. They rely on Conditional Random Fields [5] and Belief Propagation [9], Dynamic or Linear Programming [7] or Network Flow Programming [4].

Some recent tracking algorithms incorporate higher-order motion models. Many of them collapse detections from con-secutive temporal frames into single vertices. They are then used to build spatio-temporal graphs in which the motion costs are encoded either in the vertices or in the edges connecting them [2, 3]. The final trajectories are found by minimizing a cost function defined on that potentially complicated graph and that usually involves many variables. In practice, relaxation techniques are often used to speed up the optimization at the cost of not guaranteeing that the solution is the true global optimum.

In this paper, Xinchao Wang and his team introduce a network-flow Mixed Integer Programming framework that lets their model the more complex relationship between the presence of objects of a certain kind and the appearance or disappearance of objects of another kind [8]. Besids, they show that tracking interacting objects simultaneously can be achieved by modeling their motions with intertwined flow variables and by imposing linear flow constraints to enforce the fact that one object can only appear or disappear at locations where another is or has been. Their contribution is a mathematically principled approach to accounting for the relationship between flows representing the motions of different object types, especially with regard to their container or containee relationship and appearance or disappearance as well as an efficient tracklet based implementation that yields real-time performance.

2. Method of Tracking Interacting Objects

Xinchao Wang first formulates the problem of simultaneously tracking multiple instances of two kinds of objects, one of which can contain the other as a constrained Bayesian inference problem. And then team discuss the constraints and show that they result in a Mixed Integer Program (MIP).

2.1. Bayesian Inference

The team discretize the ground plane of the monitored area into a grid of L square cells which they will refer to as spatial locations. Within each one, Wang assumes that a target object can be in any one of O poses. For oriented objects such as cars, they define the pose space to be the set of regularly spaced object orientations on the ground plane; for non-oriented objects, they define the pose space to be the regularly discretized height of the ball.

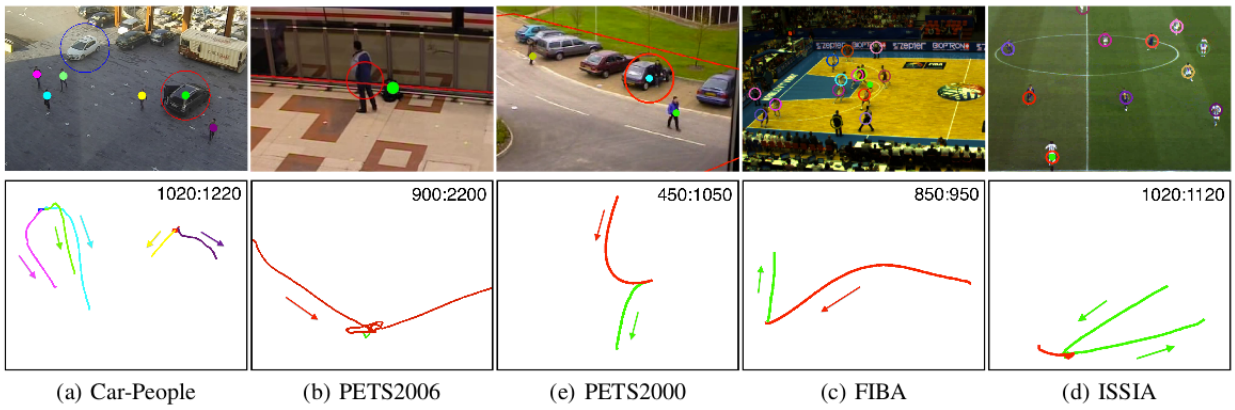

Figure 1. Tracking results on five representative subsequences taken from our datasets. Top row. Sample frames with the detected container objects highlighted with circles and containee ones with dots. Bottom Row. Corresponding color-coded top-view trajectories for interacting objects in the scene. The arrows indicate the traversal direction and the numbers on the top-right corners are the frame indices. Note that, in the FIBA case and the ISSIA case, even though there are many players in the field, we plot only two trajectories: one for the ball and the other one for the player in possession of the ball.

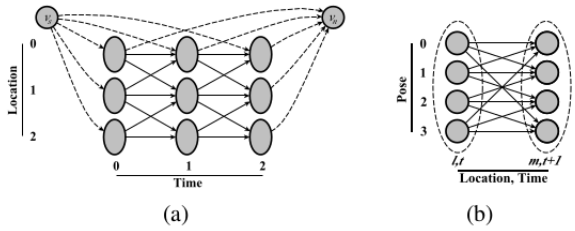

Figure 2. A graph representing three spatial locations at three consecutive times. (a) Each ellipse denotes a spatial vertex, representing a spatial location at a time instant. Some are connected to a source and a sink to allow entrances and exits. (b) Each circle inside an ellipse denotes a pose vertex, representing a pose on a spatial location or a state of an object. In this case, there are four possible poses on each spatial location.

Let k denote the state of a target object, which the team define to be the triple of location l, pose o and time t. In other words, they say an object occupies state k if it is located at l with pose o at time t. Similar to [1], which treats spatial locations as graph vertices, Wang's team build a directed acyclic graph (DAG) $G = (V, E)$ on both the locations and poses where the vertices $V = v_k$ represent the states of objects and the edges $E = e_{kj}$ represent allowable transitions between them. More specifically, an edge $e_{kj} \in E$ connects vertices v_k and v_j if and only if $j \in N(k)$. The number of vertices and edges are therefore roughly equal to OLT and $|\mathcal{N}(.)|$ OLT respectively. And they show an example of such DAG in Fig. 2.

2.2. Flow Constraints

To express all the constraints inherent to the tracking problem, Wang introduces two additional sets of binary indicator variables that describe the flow of objects between two states at consecutive time instants. More specifically, his team introduce flow variables f_{kj} and g_{kj}, which stand respectively for the number of containee and container type objects moving from state k to state $j \in N(k)$. The flow variables f_{kj} and g_{kj} are defined on the edge e_{kj} and intertwined together.

2.3. Mixed Integer Programming

The formulation defined above translates naturally into a Mixed Integer Program (MIP) with variables f_{kj}, g_{kj}, h_{lm} and the linear objective as Eq. 1:

$$\sum_{k,j \in \mathcal{N}(k)} (\alpha_k f_{kj} + \gamma_k g_{kj}). \quad (1)$$

where $\aleph_k$ and γ_k are the costs for the flow variables f_{kj} and g_{kj} respectively and they are defined as Eq. 2:

$$\alpha_k = -\log(\frac{\rho_k}{1-\rho_k}) and \quad \gamma_k = -\log(\frac{\beta_k}{1-\beta_k}). \quad (2)$$

This objective is to be minimized subject to the constraints introduced in the previous section. Since there is a deterministic relationship between the occupancy variables (x_k, y_k) and the flow variables (f_{kj}, g_{kj}).

3. Evaluation of the Method

Wang and his team tested their approach on five datasets featuring four very different scenarios that

Table 1. Comparison of SimGAN to the state-of-the-art on the MPIIGaze dataset of real eyes. The second column indicates whether the methods are trained on Real/Synthetic data. The error the is mean eye gaze estimation error in degrees. Training on refined images results in a 2.1 degree improvement, a relative 21% improvement compared to the state-of-the-art.

Sequence Name	Cameras	Frames	Locations	Containers	Containees	Container Poses	Containee Poses
Car-People Seq.0	2	350	6848	1	3	12	1
Car-People Seq.1	2	1500	6848	2	3	12	1
Car-People Seq.2	2	296	6848	2	3	12	1
Car-People Seq.3	2	2759	6848	2	8	12	1
Car-People Seq.4	2	5100	6848	4	12	12	1
PETS2006	2	3020	8800	24	1	1	2
PETS2000	1	1450	11200	3	3	12	1
FIBA	6	2850	9728	15	1	1	16
ISSIA	6	1990	34020	25	1	1	17

people and vehicles on a parking lot, people and luggage in a railway station, basketball players and the ball during a high-level competition and soccer players and the ball in a professional match. These datasets are either multi-view or monocular, and they all involve multiple people and objects interacting with each other. In Fig. 1, they show one image from each dataset with recovered trajectories and describe them above and give more details in Tab. 1.

References

[1] J. Berclaz, F. Fleuret, E. Türetken, and P. Fua. Multiple object tracking using k-shortest paths optimization. *IEEE TPAMI*, 2011. 2

[2] A. A. Butt and R. T. Collins. Multi-target tracking by lagrangian relaxation to min-cost network flow. In *CVPR*, 2013. 1

[3] V. Chari, S. Lacostejulien, I. Laptev, and J. Sivic. On pairwise costs for network flow multi-object tracking. In *CVPR*, 2015. 1

[4] A. Dehghan, Y. Tian, P. H. S. Torr, and M. Shah. Target identity-aware network flow for online multiple target tracking. In *CVPR*, 2015. 1

[5] J. D. Lafferty, A. Mccallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001. 1

[6] A. Mittal and L. S. Davis. M_2Tracker: A multi-view approach to segmenting and tracking people in a cluttered scene. *IJCV*, 2003. 1

[7] A. V. Segal and I. Reid. Latent data association: Bayesian model selection for multi-target tracking. In *ICCV*, 2013. 1

[8] X. Wang, E. Turetken, F. Fleuret, and P. Fua. Tracking interacting objects using intertwined flows. *IEEE TPAMI*, 2015. 1

[9] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Generalized belief propagation. In *NIPS*, 2000. 1