

TRACKING WITH BAYESIAN NETWORKS: EXTENSION TO ARBITRARY TOPOLOGIES

Pedro M. Jorge ^{*†} Arnaldo J. Abrantes ^{*} Jorge S. Marques [†]

^{*} ISEL, R. Conselheiro Emídio Navarro, 1950-062 Lisboa, Portugal

[†] IST/ISR, Av. Rovisco Pais, 1949-001 Lisboa, Portugal

ABSTRACT

It was recently proposed an object tracking method which is able to deal with object occlusions and group tracking, using Bayesian networks. The Bayesian network (BN) tracker has shown promising results in difficult situations but its architecture is limited to a maximum of 2 parents/2 children per node, in order to avoid the combinatorial explosion and difficult network generation procedures from the video signal. This paper addresses the major limitation of the BN tracker and presents a method to generalize the tracker to cope with arbitrary topologies, allowing the tracker to operate in more complex scenes.

1. INTRODUCTION

Object tracking is a key operation in video surveillance applications. It aims to track all the moving objects present in the scene, allowing the system to automatically follow and recognize each object and to characterize human activities.

Unfortunately, this is not an easy task, even in the case of static cameras, since the objects are often occluded by the background or by other moving objects. To solve these difficulties, several solutions have been proposed based on different types of video analysis techniques e.g., multiple hypothesis tree [1], particle filters [2], joint probabilistic data association filter [3] or heuristic algorithms [4].

Another difficulty concerns the presence of groups of people in tracking operations. This problem raises interesting challenges since it is not easy to track a person inside a group or to recover the track after the group is split. Works in this area are described in [5, 6].

We have recently proposed a tracker which is able to deal with occlusions and groups. This tracker uses Bayesian networks (BN) [7] to model the interaction among multiple trajectories and allows to correct errors when new information is retrieved from the video signal [8, 9].

Although the tracker is able to disambiguate difficult situations with occlusions and groups, the topology of the Bayesian network has to be severely restricted, in order to

keep the solution within reasonable complexity bounds. Namely, each node of the network can only have a maximum of two parents or two sons. This paper proposes a solution to overcome this difficulty and to consider more general topologies.

The paper is organized as follows. Section 2 briefly reviews the BN tracker proposed in [8]. Section 3 described the extension of this tracker to arbitrary topologies. Section 4 presents experimental results and Section 5 concludes the paper.

2. BN TRACKER

The BN tracker detects moving objects in the video signal assuming a static camera and extracts a set of object trajectories by associating regions in consecutive frames. Every time there is an ambiguity (e.g., occlusion) a new trajectory is created (see Fig. 1). Each trajectory is denoted in this context as a stroke and it may represent a single person or a group of persons.

In a second step we wish to recognize each stroke i.e., we wish to assign a label x_i which characterizes the object associated to the i -th stroke, assuming that we have observed a vector of features y_i associated to the i -th stroke. The set of all the labels associated to strokes detected before an instant t is denoted by x and the corresponding stroke features by y . Therefore $x = (x_1, \dots, x_n)$ and $y = (y_1, \dots, y_n)$, where n is the number of detected strokes.

If the stroke represents a single object, the label is an integer number. If the stroke represents a group of objects, the label is a set of integers, each one representing an object. For example, $x_i = (2, 3)$ is a group with persons 2 and 3.

The tracking problem can be formulated as follows. Given the set of observations y extracted from the video signal until time t , we wish to estimate the stroke labels x . Assuming that x, y are random variables, the most probable label assignment is given by

$$\hat{x} = \arg \max_x p(x, y) \quad (1)$$

A Bayesian network (BN) is used to model the dependence among the x_i, y_i variables; each label x_i is repre-

This work was supported by the (Portuguese) Foundation for Science and Technology (FCT) under project LTT (POS1 37844/01).

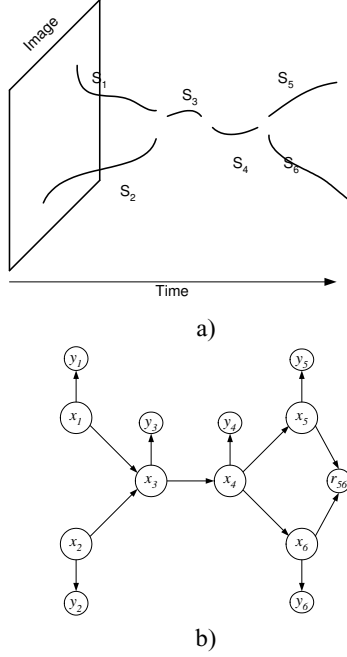


Fig. 1. BN tracker: a) Stroke detection, b) Bayesian network

sented by a node in the network which depends on a set of previous labels a_i (ancestor nodes). These dependencies account for temporal restrictions (interactions) among the strokes. The observations are also represented by nodes of the Bayesian network and each y_i depends on the corresponding stroke label x_i . Therefore,

$$p(x, y) = \prod_i p(x_i/a_i)p(y_i/x_i) \quad (2)$$

Figure 1 shows two processing stages. First a set of trajectories (strokes) is detected. Then a BN model is automatically generated. Inference is then performed using standard techniques. (r_{56} is a restriction node which guarantees that the same object does not belong to multiple trajectories after a split, see [8] for details).

The BN tracker operates as follows: the Bayesian network is automatically updated from the video signal and inference (label assignment) is periodically performed using Murphy toolbox [10]. To avoid an increase of the model complexity as time grows, only a limited number of labels (corresponding to the most recent strokes) are estimated at each instant of time. This mechanism is a way of forgetting past information which is not useful for the current decision.

The network generation involves the computation of the *network architecture, admissible labels and node distribution* from the video signal. This can be done using the procedures defined in [8] when the number of connections as-

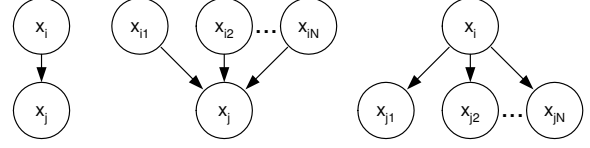


Fig. 2. Occlusion, merge and split topologies.

sociated to each node is small (maximum of 2 parents and 2 children per node). However, this is not enough to process complex interactions among different objects since it prevents the formation of groups of more than 2 persons meeting at the same time or group splitting with a simultaneous separation of many objects.

Unfortunately, the approach followed in [8] can not be easily extended to deal with these situations since it is not possible to characterize all the admissible topologies and to define a conditional probability distribution for each one of them.

A different solution for these difficulties is presented in the next section which provides algorithms for the generation of Bayesian networks with unlimited topologies.

3. EXTENSION

It is easy to deal with simple occlusions, group merges and splits with an arbitrary number of objects (see Fig. 2). The main difficulty lies in the analysis of nodes simultaneously produced by two mechanisms: merge and split (see Fig. 3a) since it is not possible to define rules for all the admissible merge-split topologies with an arbitrary number of parents/children.

To overcome this difficulty we propose to add virtual nodes between each merge-split node and the parents involved in the split (See Fig. 3b). In this way, we convert a network with an arbitrary number of local topologies into an equivalent network with only tree types of topologies: *occlusion, merge and split* (Fig. 2). Therefore only three types of rules have to be defined for label propagation and for the node probability distributions.

These rules are natural extensions of the ones used in [8] to deal with limited networks. In the case of occlusions

$$P(x_k/x_i) = \begin{cases} P_{occl} & x_k = x_i \\ P_{new} & x_k = l_{new} \end{cases} \quad (3)$$

where P_{occl} is the occlusion probability and $P_{new} = 1 - P_{occl}$ is the probability for a new label l_{new} .

In the case of a group split,

$$P(x_k/x_i) = \begin{cases} P_{split}/(2^{N_i} - 2) & x_k \in \mathcal{P}(x_i) \setminus x_i \\ P_{occl} & x_k = x_i \\ P_{new} & x_k = l_{new} \end{cases} \quad (4)$$

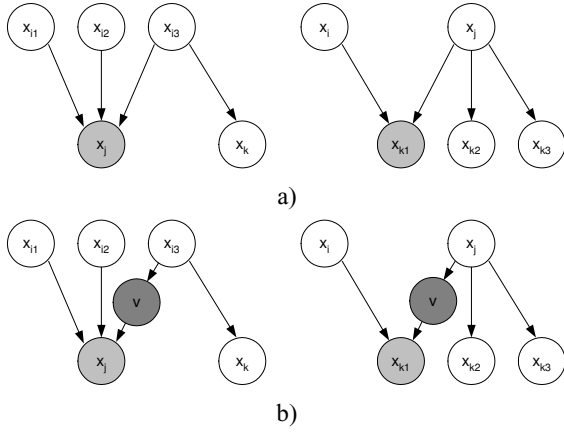


Fig. 3. a) Merge-Split topologies (light gray circles represent merge-split nodes) and b) decoupled topologies with virtual nodes (dark gray circles represent virtual nodes).

where N_i is the number of individual labels in the set x_i , P_{split} is the split probability (all subgroups are considered as equiprobable) and $\mathcal{P}(x_i)$ is the partition set of x_i .

The conditional distribution of merge nodes is

$$P(x_k/\{x_i, i \in \mathcal{I}\}) = \begin{cases} P_{occl} & x_k = x_i, i \in \mathcal{I} \\ P_{new} & x_k = l_{new} \\ P_{merge}/L & otherwise \end{cases} \quad (5)$$

where L is the number of merged groups.

The probability distribution of virtual nodes are defined in the same way as a split node.

4. RESULTS

The proposed algorithm was used to track all the moving objects in video surveillance sequences. To illustrate the performance of the algorithm we will consider a short segment of video sequence with 7 people which interact forming 6 different groups with different types of group merges, splits and occlusions. Figure 4 shows 6 frames of the video sequence with the overlaid bounding boxes, detected by background subtraction [11]. This figure shows the interaction among several pedestrians with group merging and splitting.

The low level processing detected 18 strokes which are shown in Fig. 5a. This figure, shows the evolution of the mass center (column) of each active region, as a function of time. To characterize each stroke, 3 dominant colors are extracted from the active regions associated to the stroke using a clustering algorithm. The Bayesian network extracted from the video signal has 32 nodes as shown in Fig. 6 (observations nodes y_i are not represented).

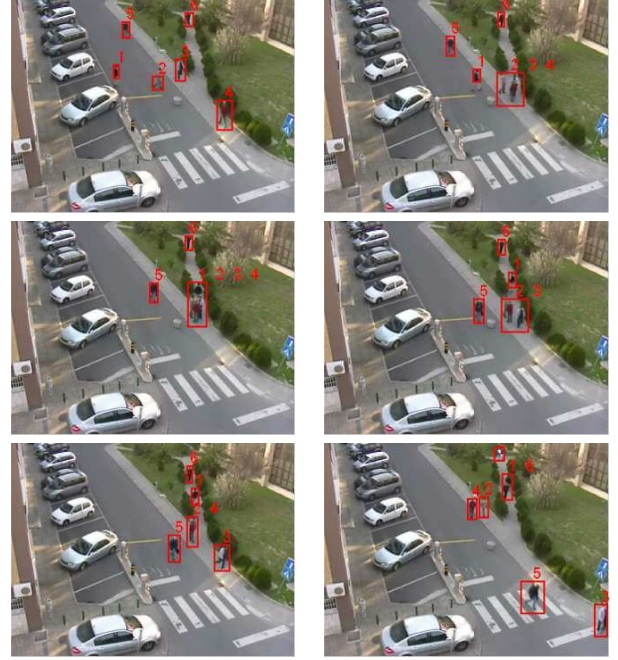


Fig. 4. Campus sequence: labeling results.

Figures 5b,4 show the output of the tracker. Figure 5b shows the labels assigned to each trajectory by the Bayesian network. Different labels are represented by different colors. We note that the algorithm manages to correctly identify each of the pedestrians which belong to the group (1, 2, 3, 4) after the group is split. Fig 4 shows the numeric labels assigned to each bounding box in the case of isolated pedestrians and groups. Labels obtained by the proposed algorithm are consistent.

The Bayesian network was automatically updated from the video signal every 5 sec. Inference results are also updated at the same rate using the Bayesian Network toolbox [10]. In order to avoid the increase of the network complexity during the experiment, only the most recent nodes are considered in the inference step. Specifically, in this example we have considered a maximum of 6 nodes from the past plus all the current strokes being followed.

The processing time associated to the network creation and update as well as periodic inference was faster than real time (73%) in a PC Centrino (1.8GHz) programmed in Matlab.

5. CONCLUSION

This paper presents an algorithm for object tracking using Bayesian networks which is able to deal with complex interactions among multiple pedestrians. The proposed algo-

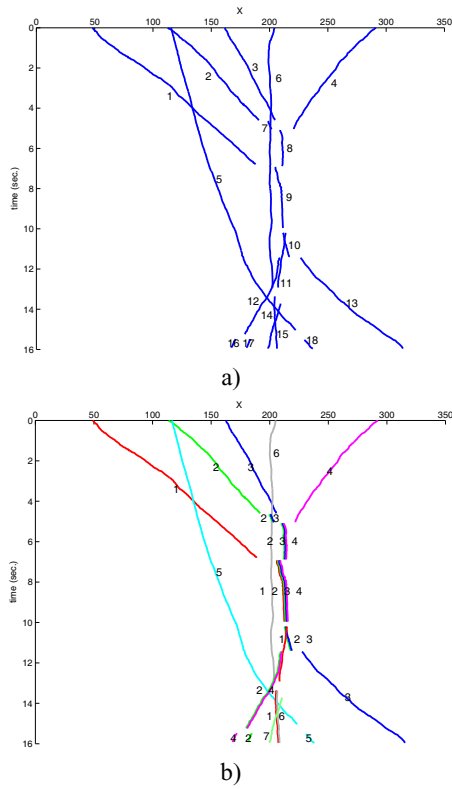


Fig. 5. a) Detected strokes and b) most probable labeling results computed with the BN tracker.

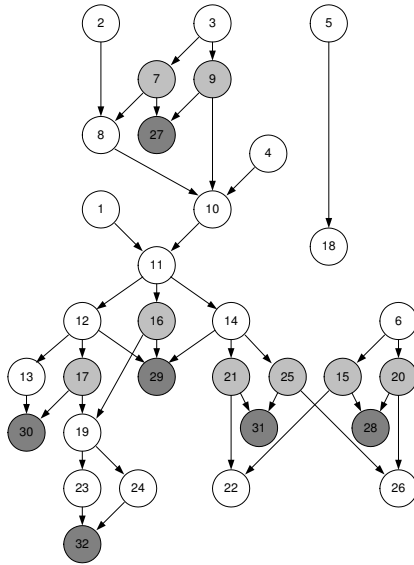


Fig. 6. The complete BN extracted from the video signal (light gray circles represent virtual nodes and dark gray restriction nodes). Observation nodes are not shown.

rithms extends the BN tracker described in [8] by allowing the use of arbitrary network topologies. Specifically, we have removed the restriction of 2 parents-2 children per node assumed in [8].

Future work should concentrate on complexity issues and the characterization of the detected strokes in the video stream which have been poorly represented by three dominant colors.

6. REFERENCES

- [1] I. Cox and S. Hingorani, "An efficient implementation of reid's multiple hypothesis tracking algorithm and its evaluation for the propose of visaul traking," *IEEE Trans. on PAMI*, vol. 18, no. 2, pp. 138–150, Feb. 1996.
- [2] K. Okuma, A. Taleghani, N. de Freitas, J.J. Little, and D.G. Lowe, "A boosted particle filter: Multitarget detection and tracking," *ECCV 2004*, vol. III, pp. 1–12, May 2004.
- [3] Y. Bar-Shalom and T. Fortmann, *Tracking and Data Association*, Academic Press, 1998.
- [4] I. Haritaoglu, D. Harwood, and L. Davis, "W4: Real-time surveillance of people and their activities," *IEEE Trans. on PAMI*, vol. 22, no. 8, pp. 809–830, Aug. 2000.
- [5] S. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler, "Tracking groups of people," *Journal of CVIU*, , no. 80, pp. 42–56, July 2000.
- [6] T. Zhao and R. Nevatia, "Tracking multiple humans in complex situations," *IEEE Trans. on PAMI*, vol. 26, no. 9, pp. 1208–1221, September 2004.
- [7] F. Jensen, *Bayesian Networks and Decision Graphs*, Springer, 2001.
- [8] P. Jorge, J. Marques, and A. Abrantes, "On-line tracking groups of pedestrians with bayesian networks," *PETS ECCV 2004*, pp. 65–72, May 2004.
- [9] P. Jorge, J. Marques, and A. Abrantes, "Estimation of the bayesian network architecture for object tracking in video sequences," *IEEE ICPR*, August 2004.
- [10] K. Murphy, "The bayes net toolbox for matlab," *Computing Science and Statistics*, vol. 33, 2001.
- [11] C. Stauffer and W. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. on PAMI*, vol. 8, no. 22, pp. 747–757, 2000.