Estimation of the Bayesian Network Architecture for Object Tracking in Video Sequences *

Pedro M. Jorge †
ISEL / ISR
pmj@isel.ipl.pt

Jorge S. Marques
IST / ISR
jsm@isr.ist.utl.pt

Arnaldo J. Abrantes †
ISEL
aja@isel.ipl.pt

Abstract

It was recently proposed the use of Bayesian networks for object tracking. Bayesian networks allow to model the interaction among detected trajectories, in order to obtain a reliable object identification in the presence of occlusions. However, the architecture of the Bayesian network has been defined using simple heuristic rules which fail in many cases. This paper addresses the above problem and presents a new method to estimate the network architecture from the video sequences using supervised learning techniques. Experimental results are presented showing that significant performance gains (increase of accuracy and decrease of complexity) are achieved by the proposed methods.

1 Introduction

Object tracking is performed in two steps in most tracking systems [1-10]. The first step tries to detect active regions corresponding to moving objects in the scene. This can be done using background subtraction [9], frame differencing or a combination of both [4]. The second stage associates active regions detected in consecutive frames and recursively computes the trajectories of the objects to be tracked. Sophisticated methods have been used to solve this problem ranging from Kalman filtering to multi hypothesis tree [5], inference methods using confidence degrees (e.g., JPDAF [2]) and particle filters [6].

A different approach is used in the Bayesian network (BN) tracker proposed in [1]. This tracker is based on the assumption that region association can be performed by simple heuristic algorithms most of the time. These algorithms can then be used to track the objects every time they appear isolated in the scene. The difficulties are usually associated with object occlusions including the superposition of several object regions and occlusions by static objects belonging to the scene (see figure 1). The low level association methods do not try to solve these difficulties and the estimated trajectories are broken in these cases. A Bayesian network is then used to link different trajectories belonging to the same object by assigning them a common label. Figure 2 shows the trajectories detected by the low level operations and the corresponding Bayesian network (see details in [1]).

Figure 1. Tracking difficulties.

The Bayesian network is automatically built during the tracking operation and tries to model causal interactions among the trajectories of moving objects. These links are then used to estimate the correct labels. Only the most important interactions should be considered in order to avoid very complex networks. This raises a difficult problem since errors in the network architecture may jeopardize the correct labelling of the detected trajectories. There is a trade off between modelling accuracy and complexity which has to be solved. The solution proposed in [1] is poor since it only considers a maximum of two parents and two sons for

*This work was supported by FEDER and FCT under project LTT (POSI 37844/01).

†These authors wish to acknowledge the support of IPL project 44/2003.

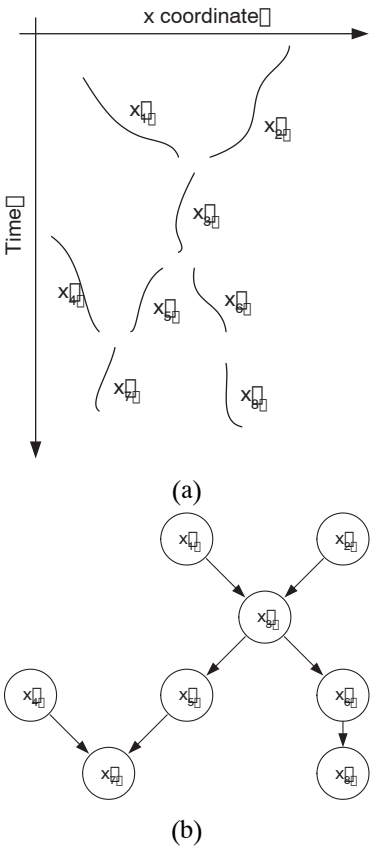

Figure 2. Detected strokes and Bayesian network.

each node, selected according to an heuristic criterion.

This paper addresses the estimation of the BN architecture from the data using supervised learning methods. The problem is formulated as a set of binary classification problems which can be solved by standard Pattern Recognition techniques. A neural network is then used to classify each admissible link as relevant or non relevant.

2 The BN Tracker

Object tracking can be split into two levels. Low level operations can be used to detect active regions in the video stream and to associate pairs of regions in consecutive frames. This can be efficiently done when each object is represented by a single active region and it is not occluded by other objects. The low level operations produce a set of trajectory segments (strokes), each of them describing the evolution of one object or a group of objects in the video stream. In general, to extract the full trajectory of each object it is necessary to link several trajectory segments. This is equivalent to a labelling operation. This problem can be solved by assigning a probabilistic label to each stroke. The interaction among different stroke labels can be modelled by a Bayesian network. The nodes of the BN are the stroke labels and the links represent the casual dependencies which are modelled by conditional probabilistic tables. The best labelling configuration can be obtained by probabilistic inference e.g., using the junction tree algorithm [8].

Each node x_i has a set of admissible labels L_i. The set of admissible labels is recursively computed taking into account the network architecture. Each node inherits the labels of its parents (see [1] for details).

The BN is defined by the graph (set of casual dependencies) and by the probabilistic model associated to each node. A critical step is the extraction of the graph from the detected strokes. A graph with many links accounts for large number of interactions among different trajectories but it leads to an intractable inference problem. Thus, a compromise between complexity and accuracy is required. We must be able to account for most of the correct interactions with a small number of links.

We wish to define a set of links (x_i, x_j) such that the network is able to represent the true labelling configuration with a complexity as low as possible. The network complexity is measured by the number of links. So we want networks with few links. However, the network architecture is used to compute the set of admissible labels L_i for each node (see [1] for details).

The network is able to correctly represent the data if the true label of each node x_i^{opt} belongs to the set of admissible labels

$$x_i^{opt} \in L_i \tag{1}$$

When this condition is false the network does not represent the data well. Let G^o be the graph with the smallest number of links which verifies (1) and d_{ij} a binary variable such that $d_{ij}^o = 1$ iff x_i is connected to x_j in G^o.

The solution proposed in [1] consists of two steps. First physical restrictions are used to define the set of admissible links. Two nodes x_i, x_j can be connected by a link if

- (causality) the stroke x_j starts after the end of x_i;
- (maximum occlusion gap) the occlusion time is smaller than T;
- (maximum velocity) the velocity during the occlusion interval is smaller than V.

However the application of the previous conditions leads to very complex networks. Therefore additional pruning conditions are applied. In fact, the networks used in [1] only allow a maximum of two parents and two sons per node. When there are more than two parents or two children, the nodes with larger occlusion gap are eliminated.

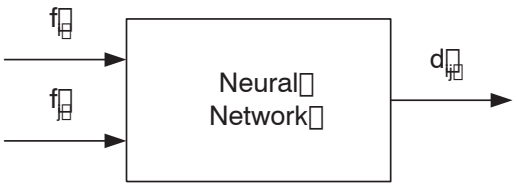

Figure 3. Neural network used to validate the link from x_i to x_j.

This approach is too restrictive since the network should be able to allow more hypothesis in complex situations e.g., when there are three or more objects being simultaneously occluded in the same region of the image plane. Furthermore, the heuristic rules should be replaced by objective criteria derived from data using learning techniques. This issue is addressed in the next section.

3 Network Pruning

Let $s_1, ..., s_N$ be the strokes detected in the video stream and $f_1, ..., f_N$ the features associated to these strokes (e.g., start time, end time, mean velocity). We would like to estimate d_{ij} from the data using the features f_i, f_j associated to each pair of strokes.

Three criteria are used to estimated d_{ij}: i) the occlusion time, ii) the occlusion distance in the image domain and iii) a criteria computed using a neural network classifier (see figure 3). The neural network tries to estimate the binary variable d_{ij} from the stroke features f_i, f_j. Two features were considered in this paper (occlusion time and occlusion distance) although this method can be easily extended to other features such as color. The neural network used in this work is a multilayer perceptron algorithm with one hidden layer (2-5-1), trained with back propagation algorithm. The cost function used to train the network is

$$J = \sum_t \sum_{(i,j)\in I} [d^o_{ij} - d_{ij}(f_i, f_j)]^2 \qquad (2)$$

where I is the set of all the links verifying the physical restrictions and t denotes time variable.

4 Experimental Results

Experimental tests were performed to evaluate the proposed methods for the estimation of the Bayesian network architecture. This involves two steps: i) training of the neural network classifier and ii) estimation of the BN architecture and computation of performance statistics. The statistics used in these tests are the probability of missing links (false negatives) and the probability of redundant links (false positives).

Two video sequences obtained with a surveillance camera in an university campus were used. The first sequence (588 s) was used to train the neural network and the other (752 s) was used for testing. Both sequences were captured at a sampling rate of 25 fps. Figure 1 shows an image extracted from the training sequence as an example.

The video sequences were first processed using low level operations described in [1] to detect all active regions and the set of object trajectories. Each trajectory is represented by a node of the BN. The optimal BN is then manually built by connecting the nodes corresponding to consecutive trajectories of the same object. This leads to a set of binary variables d^o_{ij} which defines the optimal BN architecture i.e., the network of minimal complexity which is able to represent the data. The number of nodes associated with each of the video sequences is 720 and 448, respectively.

Figure 4 shows the receiver operating curve (ROC) for the three methods described in section 3, based on the occlusion time, the occlusion distance and the neural network. The x axis displays the probability of false positives (complexity) and the y axis displays the hit probability (accuracy). The curves are obtained by varying the threshold leading to different compromises between accuracy and complexity. When the threshold is zero only physical restrictions (causality, maximum occlusion gap and maximum velocity) are applied. All the relevant links are automatically detected by the three methods but the BN has a vary high complexity. Important savings are obtained by using the three pruning techniques especially the one based on the neural network which allows a reduction of 94% of complexity with a small (1.0%) degradation of accuracy. Figure 4 also indicates the performance obtained with the method described in [1] (maximum of two parents and two sons). The method described in this paper is clearly better since it achieves higher accuracy with lower complexity if we choose an adequate threshold.

It is also interesting to study the distribution of the number of parents and sons associated with each node since this is also a measure of the network complexity. Figure 5 shows the histogram of the number of parents for a specific value of the threshold (the corresponding histogram for the number of sons is similar in this case).

5 Conclusions

Bayesian networks have been recently proposed as a tool to model the interaction among the object trajectories in tracking applications. They allow to disambiguate data conflicts arising from the superposition of different active regions (group of objects) or from occlusions. Until now, the network architecture has been defined using simple heuris-

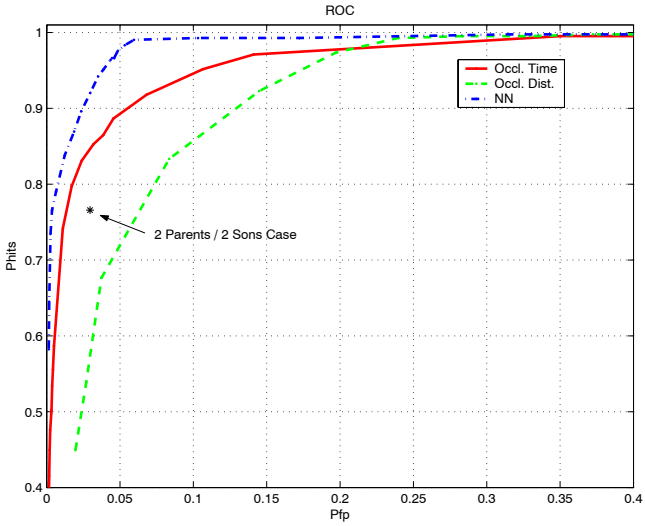

Figure 4. Receiver operating curve showing the relationship between hit probability (Phits) and probability of false positives (Pfp).

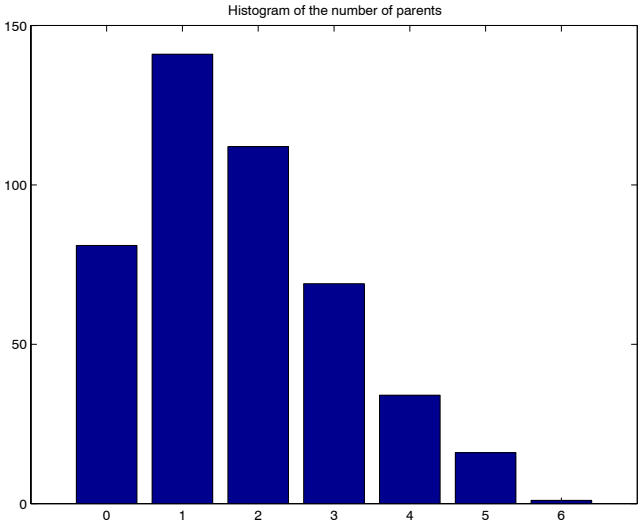

Figure 5. Histogram of the number of parents per node for the BN architecture obtained with the neural network.

tic rules which fail in several cases. This paper presents an alternative method to estimate the network architecture from the video sequences using supervised learning techniques. A neural network is trained to classify each admissible link as relevant or non relevant. This procedure allows a significant reduction of the Bayesian network complexity and an increase of the accuracy compared with previous works.

References

[1] A. Abrantes, J. Marques, and J. Lemos, "Long Term Tracking Using Bayesian Networks", IEEE Inter. Conf. on Image Processing, Vol. III, Rochester, Sept. 2002, pp. 609-612.

[2] Y. Bar-Shalom and T. Fortmann, *Tracking and Data Association*, Academic Press, 1998.

[3] I. Cohen and G. Medioni, "Detecting and Tracking Moving Objects for Video Surveillance", IEEE Proc. Computer Vision and Pattern Recognition, Fort Collins, Jun. 1999, pp. 1-7.

[4] R. Collins et al., "A System for Video Surveillance and Monitoring: VSAM Final Report", CMU technical report, 2000.

[5] I. Cox and S. Hingorani, "An Efficient Implementation of Reid's Multiple Hypothesis Traking Algorithm and Its Evaluation for the Propose of Visaul Traking", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 18, no. 2, Feb. 1996, pp. 138-150.

[6] M. Isard and A. Blake, "Condensation - Conditional Density Propagation for Visual Tracking", IEEE Inter. Journal of Computer Vision, Vol. 29(1), 1998, pp. 5-28.

[7] M. Isard and J. MacCormick, "BraMBLe: A Bayesian Multiple-Blob Tracker", IEEE Proc. 8th Int. Conf. on Computer Vision, Vol. 2, Vancouver, July 2001, pp 34-41.

[8] F. Jensen, *Bayesian Networks and Decision Graphs*, Springer, 2001.

[9] C. Stauffer and E. Grimson, "Learning Patterns of Activity Using Real-Time Tracking", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 22, no. 8, Aug. 2000, pp. 747-757.

[10] I. Haritaoglu, D. Harwood and L. Davis, "W^4: Real-Time Surveillance of People and Their Activities", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 22, no. 8, Aug. 2000, pp. 809-830.