18

Automatic Scenario Recognition for Visual-Surveillance by Combining Probabilistic Graphical Approaches

Ahmed Ziani and Cina Motamed
Laboratoire LISIC, Université du Littoral Côte d'Opale, France

1. Introduction

The online recognition and indexing of video-surveillance sequence is firstly helpful for video-surveillance operator for an on-line alarm generation by highlighting abnormal situation. The second utility concerns the off-line retrieval of specific behavior from a stored image sequence in order to discover causes of an alarm. This capability becomes naturally more powerful when the monitoring concerns a network of IP-camera over a wide area or the Internet.

The scenario recognition also known as activity recognition is an old and still active topic in computer science and several complementary approaches have been proposed by the Computer vision and the Artificial Intelligence communities. A scenario is composed on a set of elementary events linked with temporal constraints. The difficulty of human activity lies in their complexity, their spatial and temporal variability and also the uncertainty existing over the whole interpretation task. The computer vision approaches are generally focused on numerical approach by using probabilistic (Bui et al., 2004) (Hongeng et al., 2000) (Rabiner, 1989) or neural network (Howell & Buxton, 2002) approach in order to deal with uncertainty of the low level vision tasks. On the other hand, the Artificial Intelligence community has proposed more flexible symbolic approaches permitting a high level recognition capability (Tessier, 2003) (Vu et al., 2003) (Dousson & Maigat, 2007). Our main contribution in this work concerns the integration of these two complementary approaches (probabilistic and symbolic) in the global scenario recognition system.

HHMs (Hidden Markov Model) are the most popular probabilistic approach in representing dynamic systems. They have been initially used in speech recognition (Rabiner, 1989) and successfully applied over gesture or activity recognition (Starner & Pentland, 1995). An interesting feature of HMM is its time scale invariance enabling activity with various speeds. Other extensions to the basic HMM have also been used such as the Coupled Hidden Markov Models (CHMMs) for modelling human interactions (Oliver et al., 2000), and variable length Markov models (VLMMs) to locally optimize the size of behavior models (Galata et al., 2001).

However Bayesian networks have also been widely used in the computer vision community for object, event or scenario recognition. One important advantage of the Bayesian network is its ability to encode both qualitative and quantitative contextual knowledge, and their dependence.

The time aspect in probabilistic domain has led to several approaches. The first category is known as "time-slice" approach. The main technique is the Dynamic Bayesian networks (Dean & Kanazawa, 1999). DBN which can be considered as an extension of Bayesian networks, is a generalization of both Hidden Markov Models (HMM) and technique based on linear dynamical system such as Kalman filters. It assumes a Markov property by considering that a single snapshot in the past is sufficient for predicting the future.
A second category, represents the "event-based approach" also known as the "interval based approach", which allows the integration of specific nodes associated to temporal information like in (Figueroa, 1999) by the Temporal Nodes Bayesian Networks (TNBN), or Net of Irreversible Events in Discrete Time (NIEDT) (Galan & Diez, 2000). In these networks, nodes represent events that can take place at a certain time interval. A state of the node is defined as being the event outcome and the time interval at which the event is occurred.
In our context of video based scenario recognition, generally many scenarios have to be recognized, and it is important to control efficiently the system resources by using an active perception strategy (Bajcsy, 1998). In other term, the system has to focus its attention only on a set of "active scenario" with respect to the current scene behaviors.
In our opinion, in the context of scenarios based on asynchronous events, the event-based approach seems to be more appropriate and intuitive with respect to the time slice approaches. In fact, the event-based approach permits naturally to develop an active perception strategy by controlling the recognition tasks when it is necessary and in particular when a change over a node is perceived. The event based approach is also particularly relevant when the system has to manage the notion of time at several temporal granularities. In fact, the time-sliced approach adds unnecessary complexity to the network because the networks are repeated for each time slice. In the context of scenario recognition, one other main limitation of time slice approaches as DBN or standard HMM, is that there is no way to naturally represent interval-based concept: as an event e1 appears with an event e2 and finish before the end of e2. However, several extensions of HHM have been developed in order to add explicitly such time constraints. First extension concerns the hierarchical temporal structure of the HMM. In such organization, long-term layers are designed for modeling higher-level activities evolving at slower timescales. The second extension is the semi-Markov model including explicit duration HMMs. In these models, a state remains unchanged for some duration before its transition to a new state.
The proposed recognition system integrates three main layers and uses an agent based approach. This first layer in based on agents focused on basic events (Section 3). The second layer contains agents which integrate the temporal reasoning capability by using an interval based approach (section 4). The third layer combines the results of agents of the previous layers.

2. Proposed recognition algorithm

The standard approach in automatic visual surveillance is to model normal scenarios and then the system has to recognize if the current activity is normal (dangerous or safe) or unknown.
A scenario is defined by a set of events and contains in particular its start event. Each event recognition process is considered as an autonomous logical agent. A set of start event detectors are permanently in action and permits to the system to activate specific agents of the awaited scenarios. The start events are for example, "*a door is opening*", "*a car entering*", "*a pedestrian starts a specific trajectory*" *etc...*

Each specific scenario controls the execution of its useful event detectors and verifies their responses. However some detectors have the possibility to be in common with other scenarios and may be previously activated.
The partial recognition strategy based on the notion of start event brings an efficient predictive capability for the high level scenario agents in order to prepare the activation of its future awaited events.

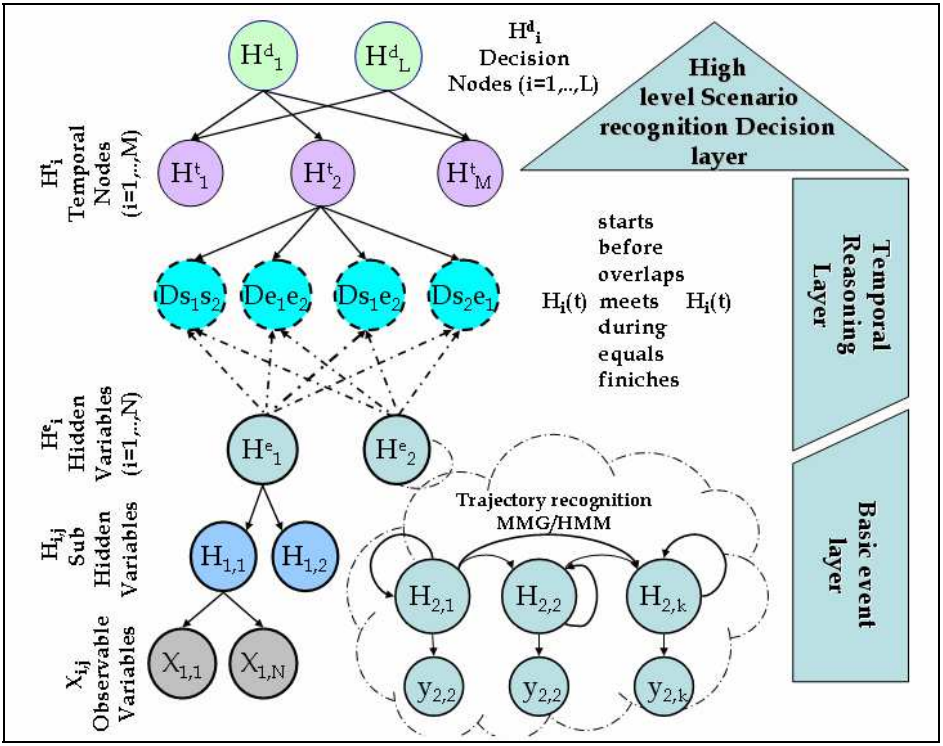


Fig. 1. Global architecture of the scenario model

The proposed approach uses probabilistic network and combines event based and HHM techniques. The system mainly takes the advantage of an event based approaches as a flexible temporal reasoning capability and it uses conventional time slice approaches for the trajectory recognition task. The global structure of the proposed network is based on the concept of the Hierarchical Bayesian Networks (fig. 1)

3. The basic events layer

3.1 Basic events with Bayesian network

The first layer the system has to recognize events as "running", "inside zone Z_i ", to detect interaction between objects as "object O_i meets objet O_j " or to recognize the trajectory of the object. Generally basic events are represented by a Bayesian network. However the trajectories recognition event is built over an hmm approach. The qualitative structure of the basic events based on Bayesian networks is defined by hand and the conditional

probabilities are adjusted easily by a learning procedure with a training data set in a supervised manner.
Generally, it is possible to learn the network parameters, from the experimental data, and in particular, the conditional probabilities parameters.

$$P\left(X_i = x_k \middle| parent(X_i) = c_j\right) = \theta_{i,j,k} = \frac{n_{i,j,k}}{\sum_k n_{i,j,k}} \tag{1}$$

Where $n_{i,j,k}$ is the count of the events in the learning database for which variable X_i is in the state x_k and its parents are in configuration c_j.
Unfortunately, in the context of visual-surveillance, it is not realistic to perform the standard learning procedure. In fact, generally it is not possible to have enough occurrences for each event. Another difficulty is that the learning process has to deal with uncertain inputs. In the presence of missing values or hidden variables, parameters for a known network structure from incomplete data can be estimated by the Expectation-Maximization (EM) algorithm.
We use the EM algorithm to learn the first layer network parameters. In order to illustrate this process, we present an example of a sub-network from an abandoned baggage scenario model used in our experiments (fig. 2). This sub-network is based on a 'naïve' Bayesian Network, and is composed by three nodes. The data collection used for learning is presented in table 1. Column "Count" shows the occurrence of a set of nodes values configuration. The value '?' represents the notion of incomplete or missing data. It highlights the situations when a node can not estimate its value. Such information is obtained by introducing a kind of self-confidence factor for the specific low-level detectors. By taking into account the incomplete data with the EM algorithm, the system naturally integrates some of the detectors limitations without reducing the quantity of learning data.

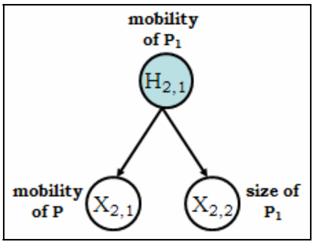


Fig. 2. An example of a naive Bayesian network

X2,1	X2,2	H2,1	Count
1	1	1	18
1	1	0	1
0	1	0	3
1	0	0	2
?	1	1	1
1	?	1	2
0	0	0	14
0	?	0	3

Table 1. A table of occurrence for the learning process

The EM algorithm is initialized with $P^{(0)}(H_{2,1}=1)=0.5$ and $P^{(0)}(H_{2,1}=0)=0.5$. Table 2 shows the evolution of the joint probabilities for the 1st and 2nd iteration. The convergence is obtained after the 13th iteration: $P^{(13)}(X_{2,1}=1|H_{2,1}=1)=0.857$ and $P^{(13)}(X_{2,2}=1|H_{2,1}=0)=0.985$. Figure 3 shows the evolution of these probabilities during the learning process.

Iterations		$X_{2,1}=0$	$X_{2,1}=1$	$X_{2,2}=0$	$X_{2,2}=1$
1	$H_{2,1}=0$	0.50	0.49	0.39	0.60
1	$H_{2,1}=1$	0.45	0.54	0.44	0.56
2	$H_{2,1}=0$	0.48	0.51	0.41	0.58
2	$H_{2,1}=1$	0.47	0.52	0.43	0.57

Table 2. Iterations of EM

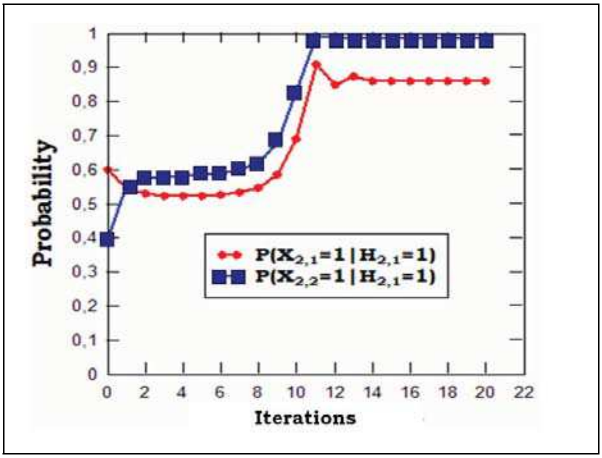


Fig. 3. Evolution of the estimated probabilities

3.2 The trajectory recognition

The trajectory recognition task represents an important feature of our scenario recognition system. It integrates a Hidden Markov Models (HMM) approach which is well adapted for sequential data recognition. The core of the proposed algorithm uses the approach of (Porikli, 2004) for modeling the trajectories and their temporal evolution. The feature vector representing a trajectory T_j is the vector containing the N successive values of object position $W=(x,y)$: $O_N = (W_1,..., W_N)$. Then a more compact representation of trajectory based on a vector quantization is built. It is based on a standard K-Means algorithm and it classifies trajectories into K clusters such that some metric relative to the centers of the clusters is minimized. The main drawback of the standard k-means concerns the number of cluster that should be known a priori. We have used an automatic estimation of this number for each trajectory model. The algorithm computes the mean length for each of trajectory collection in the image coordinate space. For a specific and controlled environment, the length estimation can be also obtained in the real coordinate by using a standard homography transformation. Then the number k of cluster is defined proportionally with

this length. This strategy permits a specific vector quantization by taking account the complexity of each trajectory model.
We have developed a predictive trajectory recognition algorithm in adequacy with the active strategy of the global scenario recognition. It permits to focus the recognition on the relevant trajectories and in the same manner to help the system to remove non plausible scenarios. The recognition is based on a recursive partial recognition of trajectory from their starts (sub-trajectory).
The HMM based recognition takes into account a set of learning data for each trajectory collection. At each instant the system uses partially these data with respect to the set of cluster that the object has crossed for each trajectory collection.
The conditional observation distribution model is based on a mixture of Gaussian (GMM). The Expectation-Maximization algorithm is used for computing the parameters of a parametric mixture model distribution GMM.
The main characteristic of our algorithm is the use a distance between current sub-trajectory and the sub-trajectories used for learning of each model of trajectory (trajectory collection). In order to compare two Hmm models of sub-trajectories λ_1 and λ_2, the similarity measure proposed by (Starner & Pentland, 1995) is used:

$$D_s(\lambda_1,\lambda_2) = \frac{1}{2}[D(\lambda_1,\lambda_2) + D(\lambda_2,\lambda_1)] \tag{2}$$

With

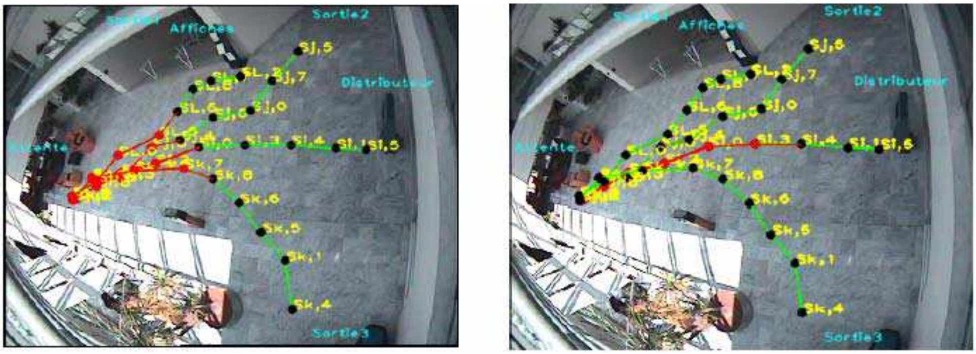
$$D(\lambda_1,\lambda_2) = \frac{1}{n_2}\left[\log P(S_{o_2} \mid \lambda_1) + \log P(S_{o_2} \mid \lambda_2)\right] \tag{3}$$

Where $S_{oj}=W1,W2,...,W_{nj}$ and $P(S_{oj} \mid \lambda_j)$ represents the probability of observation of S_{oj} by the model λ_i. n_j is the length of the trajectory λ_j (number of states).
For each $collection_i$ of trajectories (length $\{l\}$), $T_{i,k,l}(k=1,...,N)$, represents the set of HMM models linked with sub-trajectory used for learning. The algorithm estimates a distance: DM obtained by the mean of distances between current trajectory T_c and all the sub-trajectory $T_{i,k,l}$ of the $collection_i$.

$$DM(T_c, collection_i) = \frac{\sum_k D_s(T_c, T_{i,k,l})}{card(collection_i)} \tag{4}$$

The collection of trajectories which obtains the lowest distance DM with the trajectory T_c is chosen as the most likely predictive trajectory model. The figure 4 shows an illustration of trajectory prediction situation (Pets'2004 dataset). The prediction rate of the trajectory with the length $\{l\}$ is defined with the respect of the total number (n_f) of states in each collection:

$$P_{prediction}(T_c) = \frac{1}{n_f} P(S(o)|\lambda_l) \tag{5}$$



(t) (t+2)

Fig. 4. An illustration of predictive trajectory recognition at instants t and t+2

4. Temporal reasoning layer

The temporal reasoning layer uses explicitly the event-based approach. The objective of the proposed temporal layer is to evaluate temporal constraints for each event and also to estimate relations between coexisting events. The temporal constraints represent the duration of existence (relative or absolute) of an event. Relations between events reveal their mutual temporal dependency (before, after, during etc..).

The proposed temporal reasoning layer is based on a set of Bayesian networks and lies explicitly on the event-based approach. This approach is inspired by the work of (Burns & Morrison, 2003) in the context of Artificial Intelligence. It permits to incorporate temporal reasoning in a Bayesian blackboard system called *AIID*: Architecture for the Interpretation of Intelligence Data.

The temporal layer operates on the "start" and "end" time of each event (H_i from the first layer). For this, the network contains specific leaf nodes exploiting the lifespan of the events (duration of existence) estimated by the first layer. The tracking of the degree of belief of a hypothesis permits to detect its "Start" and "End" time. These time points are obtained by specific detectors operating on the temporal signal of the value of belief $H_i(t)$. In order to avoid false detection, the original signal has been filtered by a smoothing operator (fig. 5). Each start or end time is represented by a normal distribution approximated by its mean and variance values. It permits to propagate the uncertainty of these time estimations over the temporal layer. The start date (respectively the end time) is obtained once the probability of the hypothesis is higher (lower, respectively) than 80%. The variance of the time estimator is fixed off-line for each category of event and is obtained by a supervised learning procedure. It exploits N events for each category and compares ground-true duration with the results of the detector (N>10, in our experiments).

Then temporal relationships between the lifespan of events H_1 and H_2 are estimated by using a generic network structure called Temporal Relation Network TRN. This network has to check the temporal relation $H^t_{1,2}$ between two events H_1 and H_2. Its general structure is defined in figure 6. For each relation, a TRN needs four specific nodes Ds_1s_2, De_1e_2, Ds_1e_2 and Ds_2e_1 which verify the temporal positions of the two events based and their start (s_i) and end times (e_i). The signification of these nodes is detailed in the table 3 for each temporal relation. A signed distance D(A,B) between two time points A and B is computed. It uses

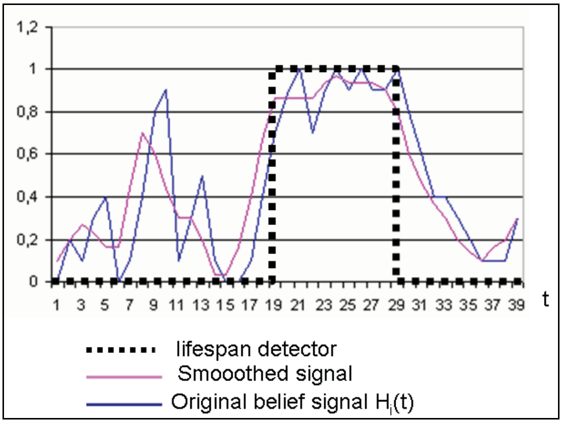


Fig. 5. Illustration of the lifespan detector based on the event belief signal

firstly the Mahalanobis distance for the normal distributions, and then the sign is obtained directly by estimating the difference between the means of the normal distributions. This signed distance permits to rank two time points by taking into account their temporal uncertainty.

In the case of the TRN "H1 equals H2" (table 3), the node Ds_1s_2 verifies the notion of start times equality of the events H_1 and H_2, (s_i = s_2) and the node De_1e_2 verifies the end times equality of the events. In this example, the node Ds_1s_2 is true when the signed distance $D(s_1,s_2)=0$.

The inference over the TRN uses explicitly an event-based approach and is activated only when one of its tracked event changes its state. The dotted lines, used in the temporal layer, represent specific links to the tracked node for the lifespan estimation.

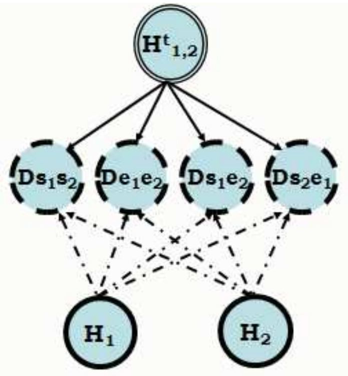


Fig. 6. The temporal relations network

The integration of the temporal constraints associated with an event, such as duration, can also be performed with the TRN presented above. It is easily obtained by linking the duration of the observed events with the constraints. Such constraint is also represented as a node and is actively controlled by the system. Generally, the constraints are defined by a relative time base with respect to other events.

Temporal Relation Network	Graphical Representation of the Relation	$D_{s_1 s_2}$ true if	$D_{e_1 e_2}$ true if	$D_{s_1 e_2}$ true if	$D_{s_2 e_1}$ true if
H_2 starts with H_1	H_1 : •—• H_2 : •——•	$D(s_1, s_2) = 0$	$D(e_1, e_2) < 0$	$D(s_1, e_2) < 0$	$D(s_2, e_1) < 0$
H_1 finishes H_2	H_1 : •—• H_2 : •——•	$D(s_1, s_2) > 0$	$D(e_1, e_2) = 0$	$D(s_1, e_2) < 0$	$D(s_2, e_1) < 0$
H_1 during H_2	H_1 : •—• H_2 : •———•	$D(s_1, s_2) > 0$	$D(e_1, e_2) < 0$	$D(s_1, e_2) < 0$	$D(s_2, e_1) < 0$
H_1 equals H_2	H_1 : •——• H_2 : •——•	$D(s_1, s_2) = 0$	$D(e_1, e_2) = 0$	$D(s_1, e_2) < 0$	$D(s_2, e_1) < 0$
H_1 meets H_2	H_1 : •—• H_2 : •—•	$D(s_1, s_2) < 0$	$D(e_1, e_2) < 0$	$D(s_1, e_2) < 0$	$D(s_2, e_1) = 0$
H_1 overlaps H_2	H_1 : •—• H_2 : •—•	$D(s_1, s_2) < 0$	$D(e_1, e_2) < 0$	$D(s_1, e_2) < 0$	$D(s_2, e_1) < 0$
H_1 before H_2	H_1 : •—• H_2 : •—•	$D(s_1, s_2) < 0$	$D(e_1, e_2) < 0$	$D(s_1, e_2) < 0$	$D(s_2, e_1) > 0$

Table 3. Temporal relationships and TRN nodes

5. Experiments

We present some results of our scenarios recognition system over two applications. The first one concerns a system for abandoned baggage scenario recognition. The second experiment is linked with an application of car park surveillance. The preliminary steps of these systems concern the motion detection, tracking of objects and their classification. The detection algorithm uses the Mixture of Gaussians in order to model the background with multiple possible states.
Our tracking algorithm basically uses a region-based approach. It uses cinematic and visual constraints for establishing correspondence. The tracking algorithm uses the Nearest Neighbor (NN) strategy. However, in the presence of merging or splitting situations, two specific procedures are launched in order to solve the association ambiguities (Motamed, 2006). The classification of object (pedestrian/car) is obtained directly by the estimation of their detection surface area in the image.

5.1 Experiment 1

The first experiment operates on the dataset delivered by the workshop PETS'2006 (PETS, 2006) (Fig. 7). It contains several challenging sequences dedicated for the performance evaluation of abandoned baggage recognition systems in the context of public transportation. The dataset is composed of multi-sensors sequences (4 overlapping cameras C1-C4) with variable complexity. All sequences are provided with information representing the calibration of camera, geometric information of the scene, and ground truth of the observed scenarios (baggage location, time of alarms). For our experiment, we have used the camera C3 which brings an interesting view of the global scene and contains less shadows artifacts. In order to compare our result with the ground truth data, a homographic transformation from the image to the ground plane is performed. The parameters of this transformation is estimated by a linear least square method by linking pairs of points from the images and their corresponding in the ground-plane.
In the figure 8, we present the model of an abandoned baggage scenario by using the proposed approach. H^t_{12} represents a temporal node verifying the relation between events

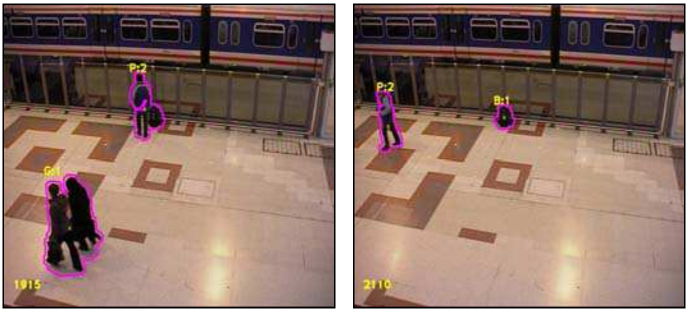


Fig. 7. Images from the dataset PETS'2006, camera C3

H_1 (detection of an object separation) and H_2 (checking an effective distant separation) The node H^t_3 in the temporal layer permits to verify a temporal duration of the node H_3 (verifying the fact that the Pedestrian P_1 goes away from the object 0_2) with a constraint node representing a duration (Δt). The lifespan of this constraint node is started after the validation of the H^t_{12} which detects the initial separation of P_1 and 0_2, this link is also drawn with dotted line.

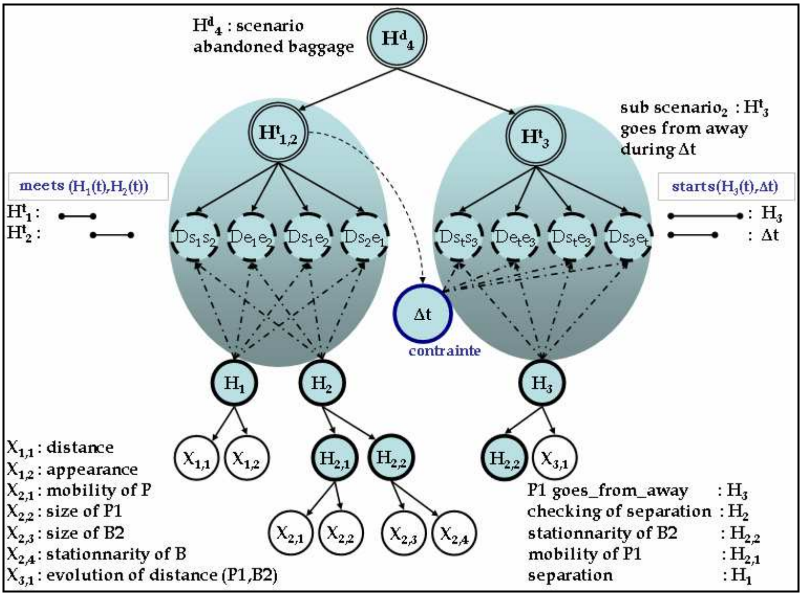


Fig. 8. Abandoned baggage model

The table 4 shows the result of our algorithm on seven sequences S1-S7. It represents the error position of the abandoned baggage, the temporal error of the alarms activation and the tracking statistics. The proposed algorithm has correctly detected the majority of the high level scenarios (6 of 7). Over the dataset S7, we can mention some image analysis errors inducing some false interpretations. This sequence contains many interacting objects involving merging situations in images. The object tracker makes few false associations and miss-classifications (2 errors on identity associations and 1 error for human/baggage classification).

Sequence	S1	S2	S3	S4	S5	S6	S7
Alarm time errors / GT(in s)	0.3	0.2	--	0,3	0.1	0.2	--
Baggage location errors / GT (in cm)	24	26	30	18	32	40	31
# True Positive alarm/ GT	1/1	1/1	0/0	1/1	1/1	1/1	0/1
# False positive alarm	0	0	0	1	0	0	1
# of tracked persons/ GT	1/1	2/2	1/1	1/1	1/1	2/2	7/6
Subjective difficulty	*	***	*	****	**	***	******

Table 4. Result of our algorithm on the PETS'2006 datasets S1-S7, camera 3

5.2 Experiment 2

The second experiment is based on two sub-scenarios linked with the entering of a pedestrian inside a car park area. Scenarios contain a set of temporal constraints and specific trajectories. The first sub-scenario verifies if the pedestrian takes a car in area P and exits the scene with the car. The second sub-scenario represents a pedestrian visiting a car with an exit from the scene without taking the car.

The figure 9 shows the global model of the car park surveillance. It contains trajectory recognition agents and temporal agents.

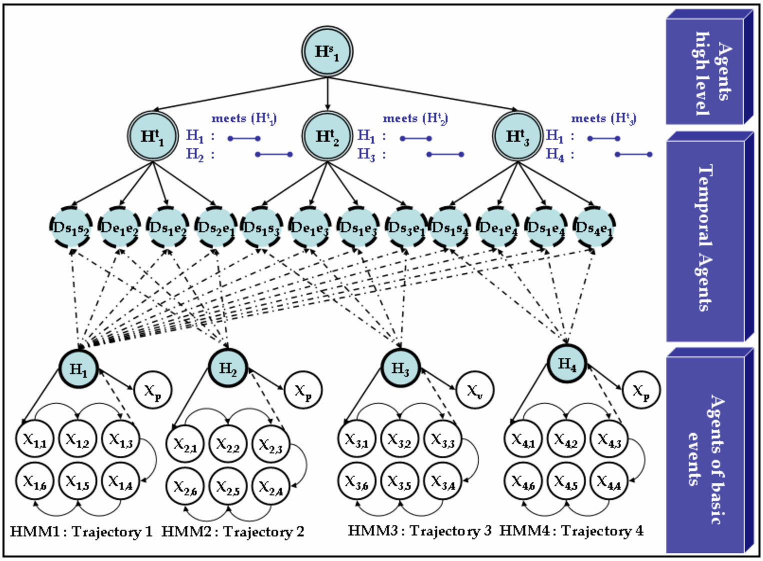


Fig. 9. Parking surveillance model

At the beginning, during the first scenario, the trajectories T_i,T_j,T_k (fig. 10:b) are partially recognized then only the trajectory T_k is confirmed (fig. 10:c). With the respect of the pedestrian paths, when the pedestrian is near the first car, the algorithm activates the second scenario. Then the second scenario as well as the first one is abandoned, because the tracked object deviates from its normal trajectories T_L or T_m (fig. 10:d). In such situation, the recognition system decides the occurrence of an atypical scenario. Fig. 11 illustrates the prediction rate of each trajectory recognition (Fig11: a, b, c, d) and its associated state evolution indicator (fig 11: e, f, g, h).

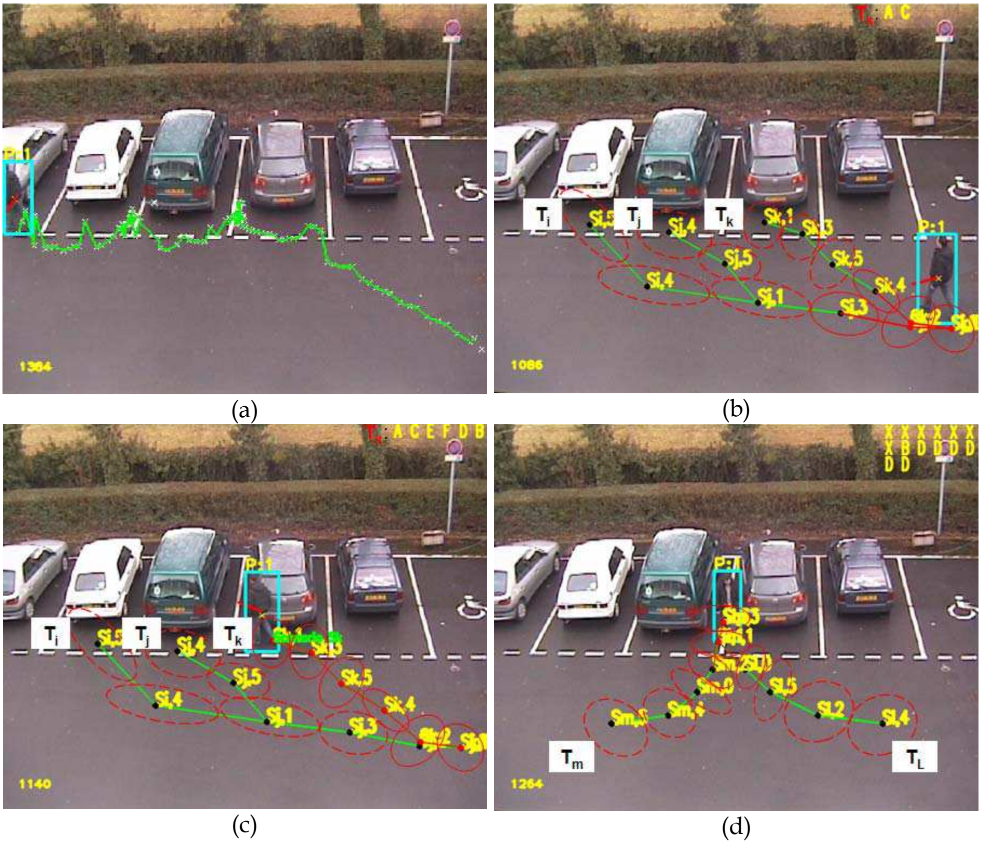
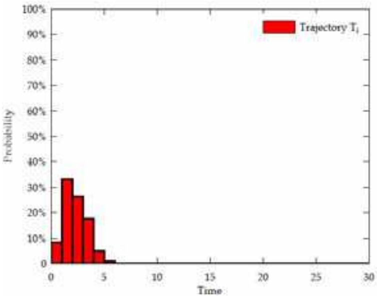


Fig. 10. (a) a global trajectory example; (b,c): recognition of trajectories T_i,T_j,T_k linked with the 1st sub-scenario; (d) : recognition of trajectories T_L, Tm linked with the 2nd sub-scenario

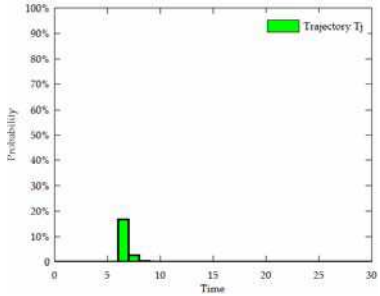
6. Conclusion

The global proposed model combines in a flexible manner, graphical probabilistic techniques in order to manage efficiently decisions uncertainty. The recognition system takes the advantage of an active perception strategy by focusing on the awaited scenarios with respect to the scene behavior. The partial recognition strategy brings efficient predictive capabilities for the high level scenario agent in order prepare the activation of its future awaited events. The first layer of the recognition permits to highlight basic events from the observed visual features. At this level, the trajectory recognition task represents an important event. We have also proposed a predictive trajectory recognition approach based on GMM-HMM model.

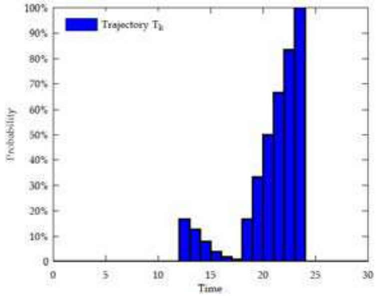
In a second layer, the use of nodes integrating temporal information in a specific Bayesian Networks, allows the temporal reasoning capabilities over the recognition task by managing various types of time constraint (qualitative, quantitative, relative and absolute). The global recognition algorithm is validated over two real world applications.



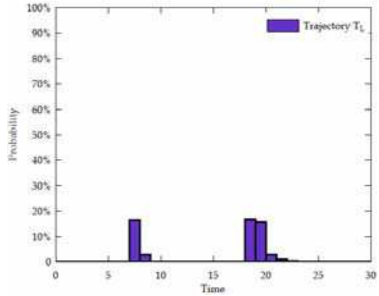
(a) Prediction rate of the trajectory Ti



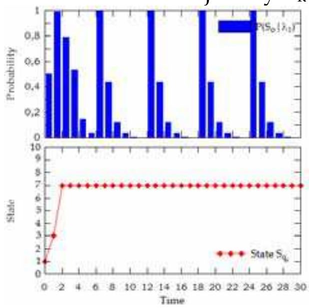
(b) Prediction rate of the trajectory T_j



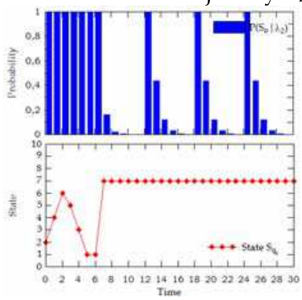
(c) Prediction rate of the trajectory T_k



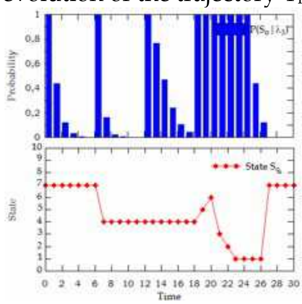
(d) Prediction rate of the trajectory T_L



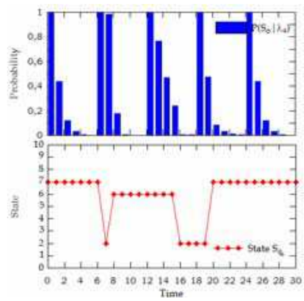
(e) State evolution of the trajectory T_i



(f) State evolution of the trajectory T_j



(g) State evolution of the trajectory T_k



(h) State evolution of the trajectory T_L

Fig. 11. Prediction of trajectories Ti, Tj, Tk, TL,: (a,b,c,d) and respectively their states evolution: (e,f,g,h)

7. References

Arroyo-Figueroa, G. (1999). A Temporal Bayesian Network for Diagnosis and Prediction. Uncertainty in artificial intelligence. Proc15th Conf Uncertainty Artificial Intelligence pp: 13-20., 1999.

Bajcsy, R. (1998). Active Perception. Proceedings of the IEEE, 76(8):996-1005, 1998.

Bui, H.H.; Phung, Q. and Venkatesh. S. (2004). Hierarchical Hidden Markov Models with General State Hierarchy. In Proceedings of the Nineteenth National Conference on Artificial Intelligence, 2004, pp: 324–329, San Jose, California,

Burns, B. & Morrison, C.T. (2003). Temporal Abstraction in Bayesian Networks. In Working Notes of AAAI Spring Symposium Workshop: Foundation and Applications of Spatio-Temporal Reasoning, 2003, pp:41-48.

Dean, T. & Kanazawa, K. (1999). A model for reasoning about persistence and causation. Computational Intelligence, 5(3), pp:142-150., 1999.

Dousson, C. & Le Maigat, P. (2007). Chronicle Recognition Improvement Using Temporal Focusing and Hierarchization. IJCAI 2007, pp: 324-329.

Galan, S.F. & Diez, F.J. (2000). Modeling dynamic causal interactions with bayesian networks: temporal noisy gates. CaNew', the 2nd InternationalWorkshop on Causal Networks held in conjunction with ECAI, pp: 1-5., 2000, Berlin, Germany.

Galata, A.; Johnson, N. & Hogg. D. (2001). Learning variable length Markov models of behaviour. Int. Journal of Comp. Vision and Image Understanding, 81(3), 2001, pp: 398–413.

Hongeng, S.; Bremond, F. & Nevatia, R. (2000). Bayesian Framework for Video Surveillance Application, ICPR00, pp: 164-170, Barcelona, Spain.

Howell, A.J. & Buxton, H. (2002). RBF Network Methods for Face Detection and Attentional Frames, Neural Processing Letters (15)(2002), pp.197-211.

Motamed, C. (2006). Motion detection and tracking using belief indicators for an automatic visual-surveillance system. Image and Vision Computing, Volume 24, Issue 11, 1, Pages 1192-1201, 2006.

Oliver, M.; Rosario, B. & Pentland. A. (2000). A Bayesian computer vision system for modeling human interactions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(8):831–843.

PETS'2006 http://www.cvg.rdg.ac.uk/PETS2006/data.html

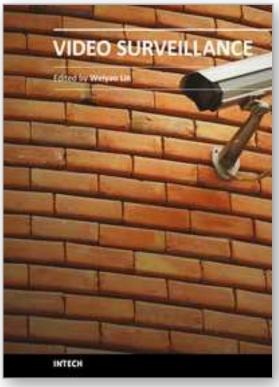
Porikli, F. (2004) Trajectory distance metric using hidden Markov model based representation. In PETS 2004 Workshop, Prague.

Rabiner. L. R. (1989). A tutorial on Hidden Markov models and selected applications in speech recognition. In Proceedings of the IEEE, volume 77, February 1989, pp 257–286.

Starner, T. & Pentland (1995) A. Real-time American Sign Language Recognition from Video Using Hidden Markov Models," Proceedings of International Symposium on Computer Vision, 1995, pp. 265-270. Miami Beach.

Tessier. C. (2003). Towards a commonsense estimator for activity tracking. In AAAI Spring Symposium, 2003, pp: 111-119, Palo Alto, CA.

Vu, T.; Bremond, F. & Thonnat, M. (2003). Automatic Video Interpretation: A Novel Algorithm for Temporal Scenario Recognition. The Eighteenth International Joint Conference on Artificial Intelligence (IJCAI'03), August 2003, pp: 9-15, Acapulco, Mexico.



Video Surveillance

Edited by Prof. Weiyao Lin

ISBN 978-953-307-436-8
Hard cover, 486 pages
Publisher InTech
Published online 03, February, 2011
Published in print edition February, 2011

This book presents the latest achievements and developments in the field of video surveillance. The chapters selected for this book comprise a cross-section of topics that reflect a variety of perspectives and disciplinary backgrounds. Besides the introduction of new achievements in video surveillance, this book also presents some good overviews of the state-of-the-art technologies as well as some interesting advanced topics related to video surveillance. Summing up the wide range of issues presented in the book, it can be addressed to a quite broad audience, including both academic researchers and practitioners in halls of industries interested in scheduling theory and its applications. I believe this book can provide a clear picture of the current research status in the area of video surveillance and can also encourage the development of new achievements in this field.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821