Management of Large Video Recordings

J.L. Patino, E. Corvee, F. Bremond, M. Thonnat

INRIA, 2004 route des Lucioles, 06902 Sophia Antipolis (FRANCE)
{jlpatino, Etienne.Corvee, Francois.Bremond, Monique.Thonnat}@sophia.inria.fr

Abstract. The management and extraction of structured knowledge from large video recordings is at the core of urban/environment planning, resource optimization. We have addressed this issue for the networks of camera deployed in two underground systems in Italy. In this paper we show how meaningful events are detected directly from the streams of video. Later in an off-line analysis we can set this information into an adequate knowledge model representation that will allow us to model behavioral activity and obtain statistics on everyday people activities in metro station. Raw data as well as on-line and off-line metadata are stored in relational databases with spatio-temporal retrieval capabilities and allow the end-user to analyse different video recording periods.

Keywords: Video understanding, activity detection, security and safety monitoring, environmental resource planning, discovery in multimedia database.

1 Introduction

The management of audio-visual streams acquired for surveillance and safety reasons is an essential point of ambient intelligence applications such as urban/environment planning, resource optimization, disabled/elderly person monitoring. In this work we have addressed the question of management and extraction of structured knowledge from large video recordings recorded over networks of cameras deployed in real sites (European project CARETAKER [1]): two different underground systems, the metro of Torino (GTT) and the metro of Roma (ATAC). Some video interpretation systems have been built in the past with similar applications. PRISMATICA [7] was a video surveillance system tested on-site in Paris and London undergrounds and able to detect overcrowding/congestion; unusual or forbidden directions of motion; intrusion; and stationarity of people. Similarly, ADVISOR [8] was tested in Brussels and Barcelona metro stations and was able to detect fighting between persons, vandalism, person jumping above a barrier, group of people blocking an exit and overcrowding situation. VISOR-BASE [9] was another video interpreting system built to store and interpret video streams from geographically distributed cameras in shopping centers and was aimed at security systems such as cashiers and entrance points monitoring. However, these systems were mainly focused on the real-time recognition of events. Recorded video contains an added value that can only be unlocked by technologies that can effectively exploit the knowledge it contains. The produced audio-visual streams, in addition to surveillance and safety issues, represent a useful source of information if stored and automatically analysed, in environment planning and resource optimisation for instance. We have thus developed techniques that automatically extract knowledge at two stages. In the first stage, events can be extracted directly from the raw data streams, such as ambient sounds, crowd density estimation, or object trajectories. The second stage of semantic information reflects relationships between tracked objects but also between tracked objects and its environment and is obtained from off-line analysis. While the second layer involves that knowledge that has previously not been modeled but discovered through unsupervised techniques and statistical analysis, the first layer corresponds to that knowledge modelled using ontologies. The ontologies describe the set of all the concepts and relations between concepts shared by the community of a given domain. An ontology is useful for experts of the application domain to use scene understanding systems in an autonomous way, to understand exactly what types of events a particular system can recognise, and for developers desiring to share and reuse activity models dedicated to the recognition of specific events. However, most of the work in ontology is dealing with structure of complex events (linguistic issues not addressing specifically video events) [10]. Several works have also addressed the limitation of standard ontologies to represent time and temporal relationships. For instance, Hobbs [11] has developed a rich ontology dedicated to time reasoning based on Allen temporal algebra [12]. A series of specific workshops sponsored by ARDA have been devoted to building ontologies of video events for video understanding applications [13]. The ontology presented in this work takes into account spatial and temporal constraints for video event recognition and interpretation.

Extracted metadata from both analysis modules, on-line and off-line, will be incorporated in knowledge management systems providing web-base content access and semantic, spatio-temporal, retrieval capabilities. For this purpose we have developed an Agent Software Methodology. The remaining of the paper is structured as follows. In section 2 we present the general architecture of the system. Section 3 introduces the ontology that has been defined for this application. The main concepts and principal results obtained from the on-line analysis are presented in section 4 while those for the off-line analysis are presented in section 5. The conclusions and some perspectives of our work are given in section 6.

2 General Architecture

Figure 1 shows the global architecture mainly composed of two different processing modules, i.e. the real-time on-line analysis subsystem, and the higher-level offline interpretation. For the storage of video streams and the metadata obtained after both, on-line video processing and off-line analysis, three different databases exist: raw database, on-line database, off-line database.

The on-line analysis subsystem takes its input directly from the data acquisition module. Streams of video are acquired at a speed of 25 frames per second. Objects and events of interest, previously defined in the ontology (see next section), are detected on real time and tracking results are written to the on-line database at a speed of 5 frames per second. Streams of video are directly written to a raw database.

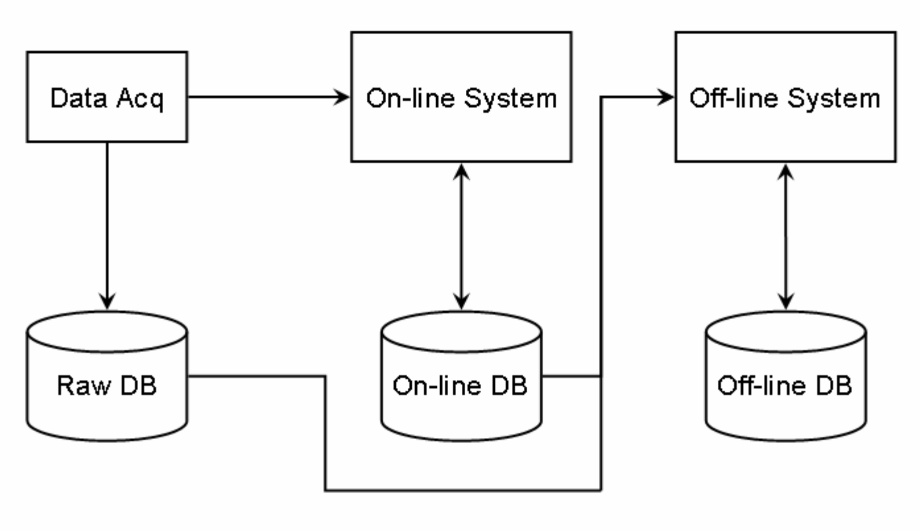


Fig. 1. General architecture

The off-line analysis subsystem takes its input principally from the on-line database as we are looking to retrieve all stored information related to a period of time that we wish to analyse. This module can also access the raw database in case the user wants to visualize a past event.

In order to allow all modules to communicate between them, we have defined, in agreement with the ontology, an exchange file format based on xml, as it has previously shown that it is an accepted standard that provides a common and understandable representation of the vocabulary, and can help to improve reusability, modularity and interoperability of the applications [2]. Large libraries of xml metadata, linked to the streams of video, are saved in both, on-line and off-line databases. With search tools, it is possible to retrieve a part of a scene, at a certain time, in the whole scene based upon the research criteria given to the search tools. In our case these are based on xml queries. Web-service technology (SOAP, WSDL, UDDI, RSS) is chosen for components and subsystems integration, because it allows reuse of high-performance interoperable components and makes the required distributed processing and communication more straightforward [3].

3 Ontologies

This section presents all the a priori knowledge and structure needed to represent video event knowledge for automatic scene interpretation. Two types of knowledge are modeled. On one side, the multi-user knowledge (safety operators, decision makers), represented by their needs, their use-case scenario definition, and their abilities at providing context description for sensory data. On the other side, the content knowledge is modeled, characterised by a first layer of primitive events that can be extracted from the raw data streams such as objects 3D dimensions or their trajectories, and a second layer of higher semantic events defined from longer term analysis and from more complex relationships between both primitive events and higher-level events. Both knowledge types are modeled through ontologies with associated semantic models and exploited in the content extraction methodologies based upon data-driven and scenario-based reasoning approaches. Extracted metadata can either be fed directly to the user for real-time information analysis or incorporated in off-line systems for statistical data analysis as described throughout this work.

There are two main types of concepts to be represented: physical objects of the observed scene, including mobile and contextual objects, and video events occurring in the scene. Terminologies describing these objects and events and terms used for scene and video analysis are listed below:

Physical object: a real world object in the scene. There are two types of physical object: physical object of interest (or mobile object) and contextual object.

Physical object of interest: a physical object evolving in the scene whose class (e.g., person, group and crowd) is predefined by end-users and whose motion cannot be foreseen using a priori information. It is usually characterized by a semantic class label, 2D or 3D features (e.g. 3D location), width and height, a posture, a trajectory, a direction, a speed, a list of objects, an initial tracking time, a reference to the camera in the scene which is best seeing it (in the case of a multiple cameras configuration), and an identifier. The identifier can either be defined locally on the current image, globally on the video sequence or globally on a scene (in a multi cameras configuration).

Contextual object: a physical object attached to the scene. The contextual object is usually static and whenever in motion, its motion can be foreseen using a priori information. For instance, the movements induced by a door, an elevator, the water coming out of a fountain, the leaves of a tree, a chair and a luggage can be foreseen.

Tracked target: corresponds to the detection and tracking of a physical object of interest. A tracked object is characterized in a scene by a unique tracking identifier.

Video event: a generic term to describe any event, action or activity happening in the scene and visually observable by cameras. Video events of interest can be either predefined by end users or learned by the system. Video events are characterized by the involved objects of interest (including contextual objects and zones of interest), their starting and ending time and by the cameras observing them. We distinguish four types of video events i.e. primitive state, composite state, primitive event and composite event which are classified into two categories i.e. state and event defined below:

+ A **state** is a spatio-temporal property of a physical object valid at a given instant or stable on a time interval. A state characterizes one or several physical objects of interest (e.g. person, crowd and vehicle) with or without respect to other physical objects.

+ A **primitive state** is a state which is directly inferred from visual attributes of physical objects computed by perceptual components. Usually, visual attributes have a numerical value and can correspond to general physical object properties for most of video understanding applications.

+ A **composite state** is a combination of states. This is the most complex granularity of states. We call **components** all the sub-states composing the state and we call **constraints** all the relations involving its components and its physical objects. For example: "Person *p1* is close to machine *m* and person *p2* stays inside zone *z*".

+ An **event** is one or several change(s) of state values at two successive time instants or on a time interval.

+ A **primitive event** is a change of primitive state values. Primitive events are more abstract than states but they represent the finest granularity of events. For example: "Person *p* moves from zone *z1* to zone *z2*".

+ A **composite event** is a combination of states and events. This is the most complex granularity of events. Usually, the most abstract composite events have a symbolical/Boolean value and are directly linked to the goals of the given application. We call **components** all the sub-states/events composing the event and we call **constraints** all the relations involving its components and its physical objects.

Five examples of video events are given below. First, an object of interest 'o' (e.g. person, group, crowd) is inside a zone 'z' if it's 3D position on the ground belongs to the polygon defining the zone (i.e. 'o IN z' is true). Second, an object of interest 'o' classified as a person is detected as close to the vending machine if this person is detected as inside the specified zone 'vending_machine_zone' and if the distance between the person and the specified equipment 'vending_machine' (i.e. 'o DISTANCE eq') is less than 1.5 meters. Third, is a mobile object 'o' is detected as staying inside a zone 'z' when the primitive state 'inside_zone(o,z)' is being detected successively for at least 30 seconds. Similarly, as fourth event, a mobile object stays at an equipment 'eq' when this object is detected successively close to the same equipment 'eq' for at least 10 seconds. The final event example corresponds to when a person is considered to be using a vending machine defined by: a mobile object is to be classified as a person and positioned within a distance from the vending machine so that the primitive state 'person_close_to_vending_machine' is detected successively for at least 10 seconds.

```
PrimitiveState inside_zone {
        PhysicalObjects: ( (o : Physical Object of Interest), (z : Zone) )
        Constraints : (o IN z) }
```

```
PrimitiveState person_close_to_vending_machine {
        PhysicalObjects: ( (o : person), (z : vending_machine_zone), (eq:
vending_machine) )
        Constraints : ( (o IN z), ( o DISTANCE eq < 1.5m )) }

PrimitiveEvent stays_inside {
        PhysicalObjects: ( (o : Physical Object Of Interest), (z : Zone) )
        Components : (c1 : PrimitiveState inside_zone(p,z))
        Constraints : DURATION(c1)>30s }

ComplexEvent stays_at {
        PhysicalObjects: ( (o : Physical Object Of Interest), (eq : Equipment) )
        Components : (c1 : PrimitiveState close_to_equipment(o,eq))
        Constraints : DURATION(c1)>10s }

ComplexEvent person_uses_vending_machine {
        PhysicalObjects: ( (o : person), (z : vending_machine_zone), (eq:
vending_machine) )
        Components : (c1 : PrimitiveState person_close_to_vending_machine
(p,z,eq) )
        Constraints : (DURATION(c1)>10s) }
```

The next section describes how object temporal information of their position in the scene allow the ontologies to detect on-line complex events from simple primitive events.

4 On-line System

The first section depicts the functionalities of the long term tracking algorithm which establish temporal links between mobile objects in order to obtain robust trajectories. The object information are then analyzed by the event detector in the second section which detect simple to more complex events based on the pre-defined ontologies.

4.1 Multiple objects tracking

Tracking several mobile objects evolving in a scene is a difficult task to perform. Motion detectors often fails in detecting accurately moving objects referred to as ‘mobiles’ which induces mistracks of the mobiles. Such errors can be caused by shadows or more importantly by static (when a mobile object is hidden by a background object) or by dynamic (when several mobiles projections onto the image plane overlap) occlusion [14].

The tracking algorithm builds a temporal graph of connected objects over time to cope with the problems encountered during tracking. The detected objects are connected between each pair of successive frames by a frame to frame (F2F) tracker [15]. The links between objects are associated with a weight (i.e. a matching likelihood) computed from three criteria: the similitude between their semantic classes, 2D dimension differences and 3D distance difference on the ground plane.

The graph of linked objects is analyzed by the tracking algorithm also referred to as the Long Term Tracker which builds paths of each mobiles according to the links established by the F2F tracker. The best path is then taken out as the trajectory of the related mobiles. Examples of tracked objects are shown in figure 2. Three major characters are evolving in this scene: two persons are one group of persons. These mobiles objects were not successively classified in all the frames due to detection errors (discussed above). However, despite the lack of well detected and classified objects, these objects were successively tracked by the long-term tracker algorithm. Tracked mobile object examples are also shown in figure 3, captured by camera number 7 in the Rome underground. It can be seen that object labeled 0 (i.e. its identifier) is being successively tracked as a person although it was sometimes misclassified. This person is shown interacting with the vending machine, standing on the left side of the image. Two other persons were also tracked, person labeled 1 and 3 which are interacting with the gates to access the train platform. Noise was also detected, such as the human activity in the office. The tracked mobile objects with the contextual information of each scene are analyzed in the section for event detection.

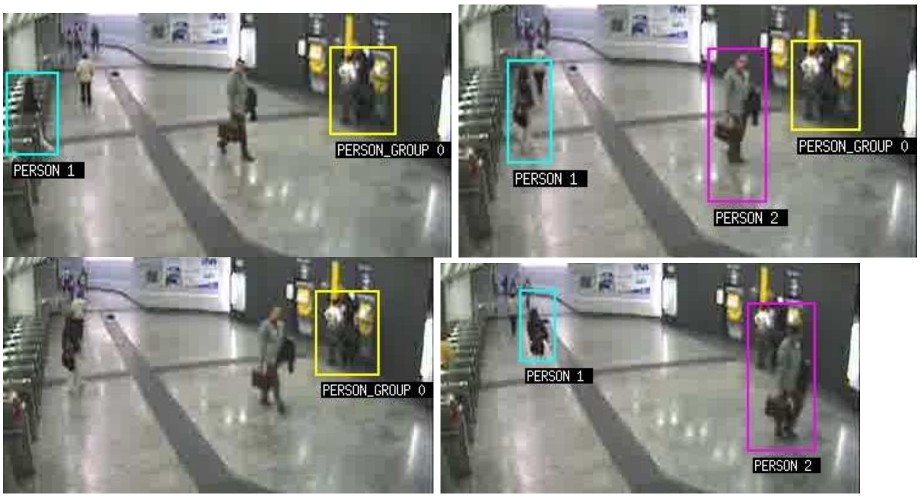


Fig. 2. Tracked objects in the Torino underground station 'Diciotto Dicembre'. Images corresponding to frames indexed 978, 999, 1008 and 1059 of a video sequence acquired at 25 fps.

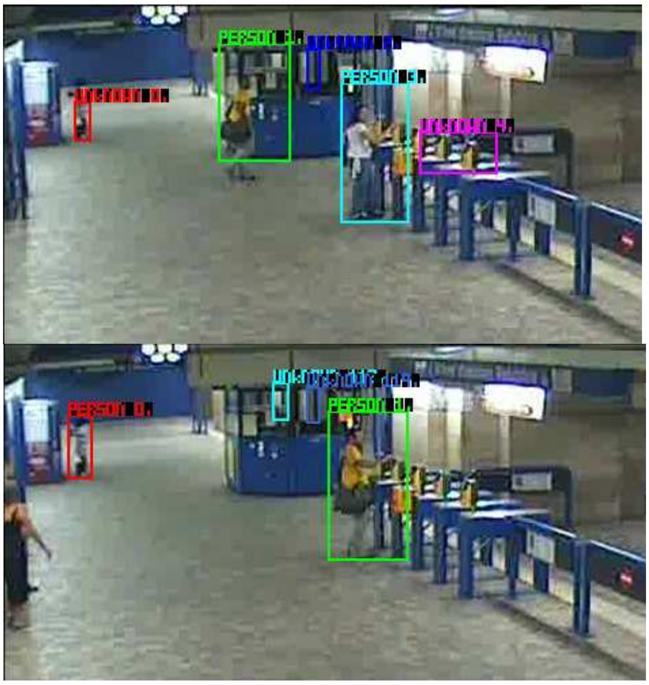


Fig. 3. Tracked objects in the Roma underground station ‘Termini’ direction ‘Rebibbia’. The 2 frames are separated by a time interval of approximately 10 seconds.

3.2 Event detection

The trajectory of each detected object given by the tracking algorithm aims to build a relationship of the objects with the contextual content of the scene. Using pre-defined ontologies and the event examples in section 2, many primitive and composite events were detected related to the detected and tracked mobile objects interacting with the scene and its content.

Figure 4 shows two events detected in the ‘Diciotto Dicembre’ station, namely the ‘stays_inside’ and ‘stays_at’ events respectively. The ‘stays_inside’ event corresponds to a group of persons being consecutively detected inside the ‘Platform’ zone for at least 30 seconds and the ‘stays_at’ event corresponds to a person being detected at gate number 7 for at least 10 seconds. 9 equipments were modeled, i.e. the 9 gates (or validating ticket machine) which allow the user to access the train, and one zone was defined in the scene: the entrance hall where people can evolve before the gates.

Figure 5 shows an example of a person using the vending machine. This person was tracked successfully for at least 10 seconds (see tracking results in figure 3) inside a small zone in front of the vending machine and close enough to it for the event ‘peson_uses_VM’ could be detected (where VM stands for vending machine). The other persons interacting in the scene were not interacting long enough with the contextual objects for any other events to be triggered (or tracks were lost due to detection errors or occlusion ambiguities).

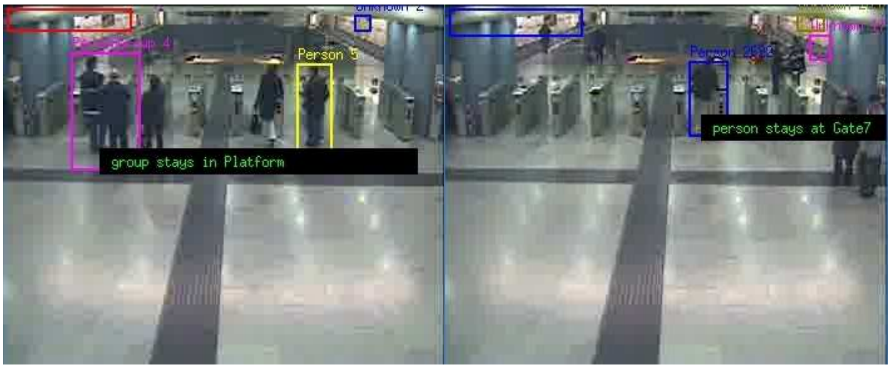


Fig. 4. Two events detected in the ‘Diciotto Dicembre’ station of Torino underground.

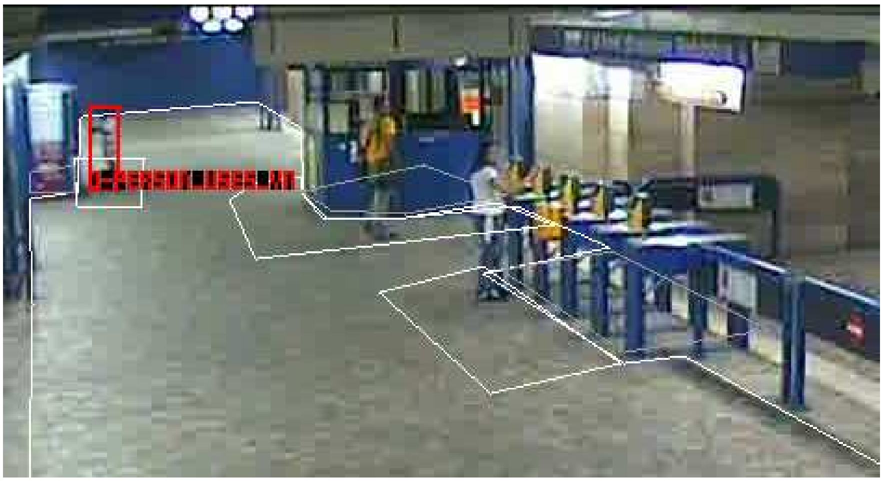


Fig. 5. Event detection in the ‘Termini’ station of the Roma underground.

Currently we are capable of detecting twelve video events. For instance, processing two hours of video from Torino metro we de-tected over 35000 events, being the most common ‘inside_zone(14486) group_inside_zone(5523), close_to_Gates(5103) stays_at_Gates(3489)’.

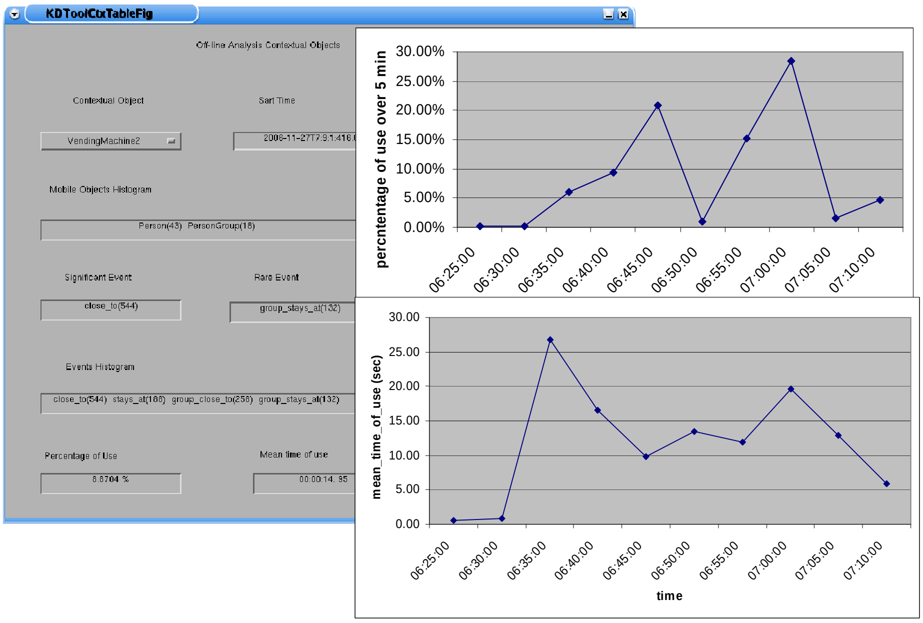
5 Off-line system

In order to have a clear and compact representation of the human activity evolving on the video and with the aim to achieve environment planning and resource optimisation, we have divided all related information to objects and events detected on the video into three different semantic tables: mobile objects table, events table and contextual objects table. Some structured knowledge representation had been introduced before [4-5], but in this contribution we propose a semantic representation which takes also into account interactions between tracked objects in the video and their environment.

Table 1. Tags included in the three different generated semantic tables

Mobile Objects Table	Events Table	Contextual Objects Table
- id. The identifier label for the object. - type. The class the object belongs to: Person, Group, Crowd or Luggage. - start. Time the object is first seen. - end. Time the object is last seen. - shape. The label describing the object's shape depending on the object's ratio height/width. - involved_events_id. All occurring Events related to the identified object. - significant_event. The most significant event among all events. This is calculated as the most frequent event related to the mobile object. - trajectory_type. The trajectory pattern characterising the object.	- id. The identifier label for the detected Event. - type. The class where the Event belongs to ('close_to', 'stays_at', …) - start. First moment on which the Event is detected. - end. Last moment on which the Event is seen. - involved_mobile_object_id. The identifier label of the object involved in that event. - involved_ctx_object_id. The name of the contextual object involved in that event.	- id. The identifier label - type. The class of the object - significant_event. The most significant event among all events but referring to contextual objects. - start; - end. refer to the first and last instant the mobile object interacts with the contextual object - involved_events_id. All occurring Events related to the identified contextual object. - rare_event. This is the rarest event. - event_histogram. Gives the frequency of occurrence of all involved events. - involved_mobile_objects_id. All detected mobile objects interacting with the contextual object of interest. - histogram_mobile_objects. Gives the frequency of appearance for all involved mobile objects. - use_duration. Percentage of occupancy (or use of a contextual object). For instance, the Ticket Machine has a 10% of use over the observation time. - mean_time_of_use. Average time of interactions between the mobile object and the contextual object.

Each column in Table 1, presented below, contains the fields that we have included for each semantic table. Apart for reordering the information in agreement with our semantic representation, there are a series of new fields we calculate in order to extract new information. With the mobile objects table we are looking to characterize the underground users. Off-line we calculate the shape, significant event and trajectory type of mobile objects. The first field allows us to estimate the number of people in a group or a crowd. The second and third fields gives us behavioral information: what is the most frequent event and what trajectory do people usually take. To describe this last field, we implemented at this point of the processing a hierarchical clustering algorithm [6] to group similar trajectories after a given observation time. The dendrogram, resulting after applying the algorithm, is unique but the final number of clusters in which the data set is to be divided is subjective. In our case, the end-user can interactively choose the final number of clusters.



KDToolCtxTableFig

Off-line Analysis Contextual Objects

Contextual Object: VendingMachine2

Sart Time: 2006-11-27T7:9:1:416.(

Mobile Objects Histogram: Person(43) PersonGroup(18)

Significant Event: close_to(544)

Rare Event: group_stays_at(132)

Events Histogram: close_to(544) stays_at(166) group_close_to(256) group_stays_at(132)

Percentage of Use: 6.8704 %

Mean time of use: 00:00:14. 95

Fig. 6. Statistics calculated for the contextual object VendingMachine2.

With the events table, we want to deduce what are the events that normally occur in the underground stations. Both, the mobile objects table and the Events table allows us to generate the contextual objects table, which is a major source of information to the underground manager for safety and resource monitoring and action planning. We have developed a graphical off-line analysis tool where the end users connect to the on-line database as shown in Figure 1 and select a period of recording time, which they want to monitor. Figure 6 shows from this graphical interface the information related to the contextual object VendingMachine2 (From the scene observed in Figure 2). The recording started at 7:09 and lasted 45 min (not shown). In this period 43 persons and 18 groups came to the Vending machine. Among all related events (shown in the Events Histogram), 'group stays at VendingMachine2' was the most rare Event meaning that users do not tend to spend long periods of time while buying a ticket during rush hours. The two last fields give the percentage of use and the mean time of use for the contextual object. For the VendingMachine2, from the 45 min of observation, only 8.8% of the time was in use and the mean time a user spends on the machine is about 23 s. In the foreground of the figure, we have the evolution on the mean time of use and the percentage of use. Every five minutes and for the whole observation time these two parameters are calculated. As seen from these graphs, people tend to spend more time in the machine when their global use is relatively low (off-peak hours). The other VendingMachine also present in the hall indicates a similar users behavior. The mean time spent by a person on the machine was 30 s, and the machine was in use 7.75 % of the observation time. The tracking results indicate that the number of persons and group of persons that came to this machine was of 30 and 10 respectively being also the most rare event associated to this machine 'group stays at VendingMachine1'.

All this information allows the underground manager to optimize the use of the stations. Three xml files are generated, one per each semantic table, and stored in the off-line database, either for further analysis or for subsequent queries.

6 Conclusions

In this paper, we have presented the methodology to manage and extract structured knowledge from large video recordings, which in this application correspond to two different underground network of cameras. From the multi-user knowledge, we have defined a specific ontology that we use to detect primitive events, then from a longer time of analysis, but always on-line, we can deduce more complex or composite events. Overall we are able to detect 12 different kinds of events directly from the streams of video. Off-line, we can further analyze the metadata associated to the detected objects and events of interest. Even if some vision errors still remain, pertinent statistics can be computed. In particular, we have analyzed the interaction between people and contextual objects. Among others, we are able to inform on the number of people on the scene, the percentage of use of the different contextual objects and the time a user spends with them. This is a major source of information for the underground manager as he can better monitor and plan the resources. All raw data and metadata are stored in separate databases for better management and we have implemented an exchange format based on xml, which also support queries with web service technologies. In the future we plan to extend the ontology to increase the number of types of events we can detect and we will also look to refine the off-line analysis such as subcategories in the undertaken trajectories to give more detailed information to the end-user for better environmental planning. We also plan to develop more advanced tools to better explore the knowledge database using data-mining techniques such as relational analysis.

7 Bibliography

[1] Carincotte, C., Desurmont, X., Ravera, B., Bremond, F., Orwell, J., Velastin, S.A., Odobez, J.M., Corbucci, B., Palo, J., Cernocky, J.: Toward generic intelligent knowledge extraction from video and audio: the EU-funded CARETAKER project. In: The IET conference on Imaging for Crime Detection and Prevention (ICDP 2006), London, Great Britain, June 13-14, (2006) 470-476

[2] Martinez, A., de la Fuente, P., Dimitriadis Y.: An XML-based representation of collaborative interactions. In: B.Wasson, S. Ludvigsen & U. Hoppe (Eds.): Computer Support for Collaborative Learning: Designing for Change in Networked Learning Environments, (CSCL 2003), Bergen, Norway, (2003) 379-384

[3] Lienard, B., Hubaux, A., Carincotte, C., Desurmont, X., Barrie, B.: On the Use of Real-Time Agents in Distributed Video Analysis Systems. In: IS&T/SPIE 19th Annual Symposium on Electronic Imaging, San Jose, California USA, January 28/February 1 2007.

[4] Lin, H., Chen, A.L.P.: Motion event derivation and query language for video databases. In: Proceedings of SPIE, Vol. 4315 (2001) 208-218

[5] Liu, D., Hughes, C.E.: Deducing Behaviors from Primitive Movement Attributes. In: Defense and Security Symposium, Proceedings of the SPIE, Vol. 5812 (2005) 180-189

[6] Kaufman, L., Rousseeuw, J.P.: Finding groups in data, Wiley-Interscience (1990)

[7] Velastin, S.A., Boghossian, B.A., Lai Lo, B. P., Sun, J., Vicencio-Silva M.A.: PRISMATICA: Toward Ambient Intelligence in Public Transport Environments. IEEE Trans Syst Man Cy A. 35 (2005) 164-182

[8] Cupillard, F., Bremond, F, Thonnat, M.: Video understanding for metro surveillance. In: Proceedings of the IEEE International Conference on Networking, Sensing and Control, special session on Intelligent Transportation Systems (IC-NSC), Taipei, Taiwan (2004)

[9] Piater, J., Richetto, S., Crowley, J.: Event based activity analysis in live video using a generic object tracker. In: Freyman, J. (ed.) Proceedings of the 3rd IEEE Workshop on performance evaluation of tracking and surveillance (PETS), Copenhagen, Denmark (2002)

[10] Narayanan, S.: KARMA: Knowledge based Actions Representations for Metaphor and Aspect, PhD Dissertation, University of California at Berkeley, CA, USA (1997)

[11] Hobbs, J.: A DAML Ontology of Time, http://www.cs.rochester.edu/~ferguson/daml/

[12] Allem, J., Ferguson, G.: Actions and Events in Interval Temporal Logic. In: Stock, O. (ed.) Spatial and Temporal Reasoning, Kluwer Academic Publishers (1997) 205 – 245

[13] Bremond, F., Maillot, N., Thonnat, M., Vu, T.: Ontologies For Video Event, Technical report INRIA Sophia Antipolis no. 5189 (2004)

[14] Georis, B., Bremond, F., Thonnat, M., Macq, B.: Use of an Evaluation and Diagnosis Method to Improve Tracking Performances. In: Proceedings of the 3rd IASTED International Conference on Visualization, Imaging and Image Proceeding (VIIP) vol. 2 (2003)

[15] Avanzi, A., Bremond, F., Thonnat, M.: Tracking Multiple Individuals for Video Communication. In: Proceedings of the IEEE International Conference on Image Processing, vol 2 (2001) 379-382