MTAP manuscript No.
(will be inserted by the editor)

Video Surveillance Online Repository (ViSOR): an integrated framework

Roberto Vezzani · Rita Cucchiara

Received: date / Accepted: date

Abstract The availability of new techniques and tools for Video Surveillance and the capability of storing huge amounts of visual data acquired by hundreds of cameras every day call for a convergence between pattern recognition, computer vision and multimedia paradigms. A clear need for this convergence is shown by new research projects which attempt to exploit both ontology-based retrieval and video analysis techniques also in the field of surveillance. This paper presents the ViSOR (Video Surveillance Online Repository) framework, designed with the aim of establishing an open platform for collecting, annotating, retrieving, and sharing surveillance videos, as well as evaluating the performance of automatic surveillance systems. Annotations are based on a reference ontology which has been defined integrating hundreds of concepts, some of them coming from the LSCOM and MediaMill ontologies. A new annotation classification schema is also provided, which is aimed at identifying the spatial, temporal and domain detail level used. The ViSOR web interface allows video browsing, querying by annotated concepts or by keywords, compressed video previewing,

Imagelab, Dipartimento di Ingegneria dell'Informazione.
University of Modena and Reggio Emilia, Italy.
E-mail: {roberto.vezzani, rita.cucchiara}@unimore.it

media downloading and uploading. Finally, ViSOR includes a performance evaluation desk which can be used to compare different annotations.

Keywords Video repository · video surveillance · annotation · ViSOR

1 Introduction

Video Surveillance is nowadays a well established discipline which joins pattern recognition and computer vision communities for the extraction of semantically valuable knowledge from videos related with security and safety issues. The availability of techniques and tools and the capability of storing huge amounts of visual data acquired by hundreds of cameras every day call for a convergence between pattern recognition, computer vision, and multimedia paradigms.

A clear need for this convergence is shown by new research projects, which attempt to exploit both ontology-based retrieval and video analysis techniques also in the field of surveillance.

Recently, a survey on the technologies used for managing video surveillance data has been conducted by means of the VIDI-Video [1] project. An excerpt of some questions and the related answers is reported in Fig. 1. The questionnaire was basically conceived to highlight the inadequacy of traditional free text annotation and query systems applied to surveillance. Looking at the reported results, it seems clear the video surveillance community needs new concept-based technologies. In particular, even most interviewees perform event, object and people detection, only few people use a standard schema, ontology or even controlled lexicon to annotate videos. Thus, queries by concept (that are desirable by more than half of the users) are not possible.

In addition, all researchers working in video surveillance and computer vision lack common large benchmark suites with annotated videos and ground truth data to provide fair performance evaluation and open discussions about techniques and methodologies.

A preliminary work has been presented in [3], while this paper gathers and describes all project-related aspects and the system developed. In brief, the key features offered by ViSOR are:

- creation and updating of a wide ontology of concepts related to surveillance and security
- availability of an extensive repository of surveillance videos for research, analysis and benchmarking
- video acquisition for scientific purposes and with consenting actors, in full compliance with privacy legislation

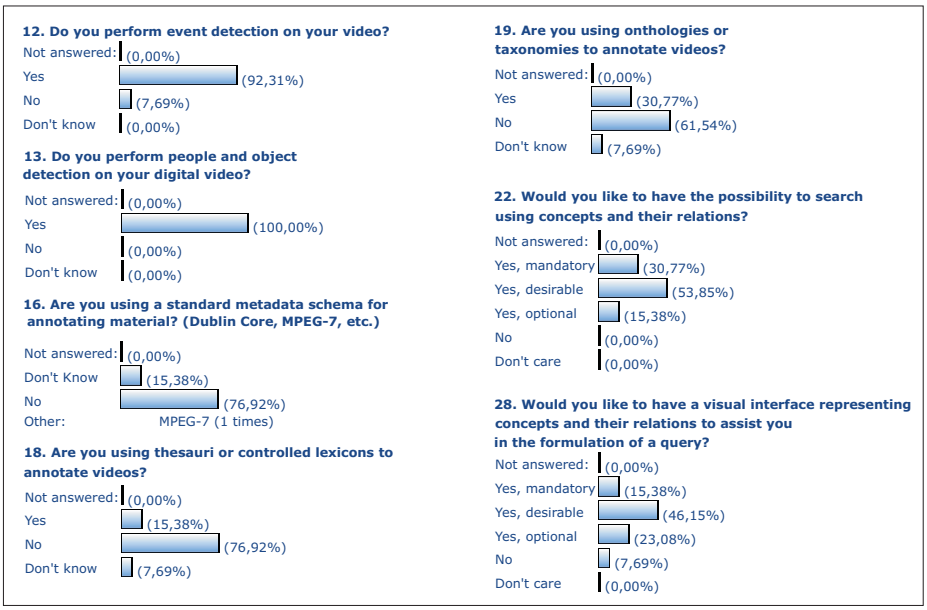


Fig. 1 VIDI-Video survey excerpt; for the complete results see [2]

- videos enrichement with both manual and automatic annotations
- availability of tools for manual and automatic annotation
- ViSOR is OPEN and FREE, and users are allowed to download and upload their own videos and annotations
- differently from multimedia video archives, specific requirements for surveillance applications are taken into account; for example, it is possible to share camera calibration data.

This paper is organized as follows: available datasets for surveillance are reported in Section 2, highlighting some limits that we aim to overcome with ViSOR. The whole ViSOR framework and the web interface are described in Section 3. Section 4 proposes the ViSOR concept list for the surveillance domain, while Section 5 reports a new annotation classification. Section 6 shows the different annotation formats supported by the system. The performance evaluation desk is described in Section 7.

2 Benchmarks and video surveillance datasets: a short review

Academic and industrial researchers make intensive use of benchmarks and common data sets. In many other research fields large databases are available for scientists and companies, such as FERET (FacE REcognition Technology) for biometry [4], TREC (Text REtrieval Conference) for text, or TRECVid for video retrieval [5]. In addition, corresponding ontologies have been defined in several cases for enhanced formalization and description of the data stored. TRECVid, for example, exploits the LSCOM (Large-Scale Concept Ontology for Multimedia) concept list [6].

In video surveillance, many performance evaluation projects are increasing their popularity using some tools and available datasets. Some open source tools, such as

ViPER-GT and ViPER-PE [7] constitute an inter-operable platform to manually select concepts and events in videos, generate ground truth and annotate videos into XML files. The ViPER annotation format [8] is widely exploited by available vide databases, which are created in workshop and conference contests, like the PETS workshop series [9] or the VSSN workshops of the ACM Multimedia Conference [10] and in those environments that become available from some European or national projects such as I-Lids [11] and Etiseo [12]. These contests propose their own datasets. Some examples are reported in Table 1 and illustrated with a sample frame in Fig. 2; most of them contain few videos only; just TRECVid now proposes 100 hours of surveillance videos, but it is not freely available.

However, most of them have some drawbacks. The first is often the lack of generality and their narrow focus on few specific problems of computer vision and pattern recognition. PETS, for example, is the most important project that has somehow contributed to start research in surveillance and has been deeply exploited in many research activities; nevertheless, it has been proposed with a specific annotation and for a specific task.

The second limit is the lack of user interaction; often users cannot share their videos and annotations, nor provide useful comments and requirements. Moreover, the ontology defined usually is not available, and there is no graphical tool or querying system to select only the subset of videos useful for a given application.

3 The ViSOR framework

ViSOR is not only a web archive for videos and annotations, but also a wider framework composed by different entities among which the user community plays an undoubted role. The main idea is to exploit the collaborative paradigm of Web 2.0 community, to

Dataset		Topics	Size
1.BEHAVE [13]		Unusual activities	8 with ground truth
2.CANDELA [14]		Indoor left-luggage and traffic monitoring on road intersection	16 indoor
3.CAVIAR [15]		Different scenarios of interest. These include people walking alone, meeting with others, window shopping, entering and exiting shops, fighting and passing out and last, but not least, leaving a package in a public place	60 videos
4.Etiseo [12]		Object Detection, Object Localization, Object Tracking, Object Classification.	86 video clips
5.i-Lids (AVSS 2007) [11]		Stopped vehicles and abandonded luggage	14 sequences
6.ObjectVideo Virtual Video [16]		Tool to generate virtual video sequences for surveillance purposes.	-
PETS [9]	7.2001	Outdoor people and vehicle tracking	5 sequences
	8.2002	Indoor people tracking (and counting)	6 sequences
	9.2004	Use of CAVIAR dataset - People tracking and activity recognition	28 sequences, 6 scenarios
	10.2006	Surveillance of public spaces, detection of left luggage	7 datasets (4 camera views each one)
	11.2007	Multisensor sequences containing loitering, attended luggage removal (theft), and unattended luggage	8 datasets (4 camera views each one)
12.SELCAT [17]		Level crossing monitoring for detection of stationary vehicles.	8 sequences
13.SPEVI [18]		Face detection and tracking	10 sequences
14.Traffic datasets by Institut für Algorithmen und Kognitive Systeme [19]		Traffic surveillance, in particular at road intersections	14 sequences
15.ViSOR		Indoor and outdoor surveillance sequences; annotation data for object detection, tracking, events, and much more.	162 sequences at 01/07/2008 (in progress)
16.VSSN [10]		background subtraction competition	7 sequences (now included in ViSOR)
17.PERCEPTION group Multiple-Camera Database [20]		Multiview dataset for view-invariant human action recognition	13 daily-live motions performed each 3 times by 11 actors
18.TRECVid 2008 [21]		Gatwick Airport surveillance video data (courtesy of the UK Home Office).	100 hours (10 days * 2 hours/day * 5 cameras)
19.CMU Motion Capture Database [22]		Trials of Human Interaction, Interaction with Environment, Locomotion, Physical Activities, Sports, and others.	There are 2605 trials in 6 categories and 23 subcategories
20.HumanEva [23]		Calibrated video sequences synchronized with 3D body poses obtained from a motion capture system. The database contains subjects performing 6 common actions (e.g. walking, jogging, gesturing, etc.).	7 calibrated video sequences (4 gray scale and 3 color), 4 subjects, 6 actions

Table 1 Existing datasets

Fig. 2 Sample frames for the datasets of Table 1

bring together the ontology-based annotation and retrieval concepts and the requirements of computer vision and video surveillance communities.

Surveillance videos, together with their metadata annotations, are key aspects of ViSOR, but other tools such as *Online Performance Evaluation*, *Forum*, *Systems and Scientific Paper databases* enrich the framework. In the following sections each block will be described in detail.

From a functional point of view, the ViSOR framework can be illustrated as shown in Figure 3.

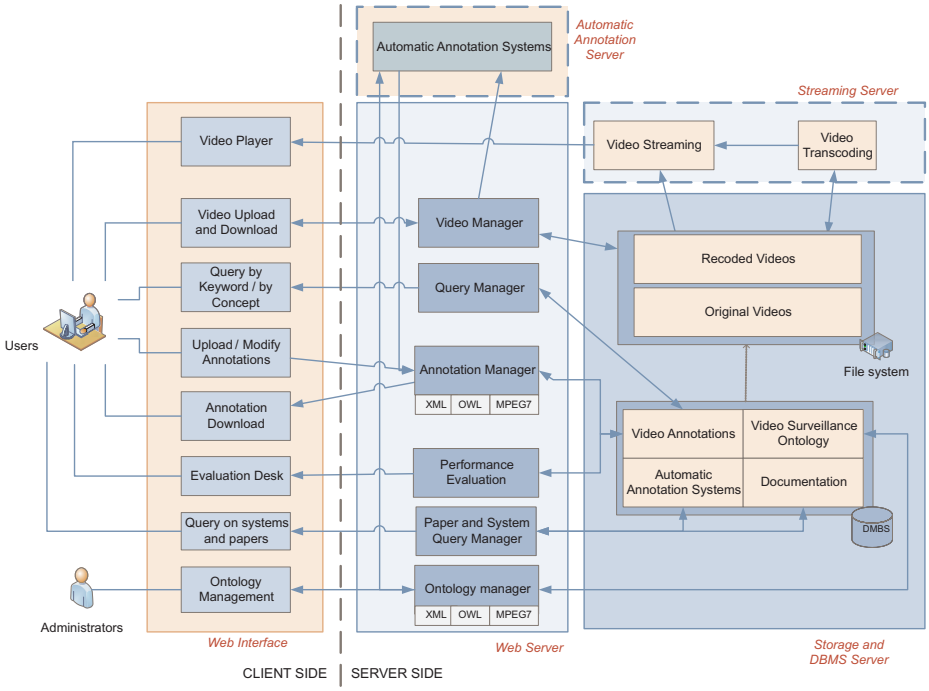


Fig. 3 Functional schema of the ViSOR framework

The system has been conceived as a web application; we can therefore highlight the client and the server side of the framework. The former contains all the functionalities which should be available by both normal and administrator users. In particular, ontology management is reserved to administrators only, while the normal user can browse, show, download, and also upload videos and annotations. All these tasks exploit software components available through a dedicated web server. In the depicted schema we have split the server side into four logical items. The *Storage and DBMS server* is the ViSOR core and contains videos in a file system organization and annotations in a DBMS. In addition, the same DBMS stores the other ViSOR data, such as the ontology, a bibliography of important surveillance scientific work, and a collection of system and tool descriptions and references added by the users. The video storage subsystem contains the related transcoding versions, in order to provide the users with

suitable video formats without a real time recoding. For example, an MPEG1 version and a flash compressed preview version are always created and stored. Anyway, some applications and the ViSOR web interface itself can make use of a video streaming technology instead of a traditional file delivery. Subsequently, a video streaming server has been installed and enhanced with on-the-fly video transcoding modules.

The annotations are stored in a DBMS. Finally, ViSOR has been enriched with an Automatic Annotation Server, which automatically extracts information from the videos uploaded and generates automatic annotations. More precisely, a battery of standalone applications are running on this server, each one interacting with the storage and DBMS server by means of the Web Server components. This way, we can avoid security issues as well as inconsistency problems: there is a single access point for generation of both manual and automatic annotations. Moreover, this solution allows third parties to develop their own annotation tools. Each application can be actually dedicated to a subset of ViSOR concepts.

The ViSOR web interface has been designed in order to share videos and annotation contents. Figure 3 shows the main modules of the ViSOR web interface. In particular, the web interface allows the users to: (i) upload videos, download them exploiting a visual browse interface or a query by keywords; (ii) download, upload or modify annotation data relating to a video; (iii) retrieve videos by concepts, which means being able to look for the desired concept within the annotation database and get a list of annotations and the related videos containing that concept.

A screenshot of the video section as shown by the ViSOR web interface is reported in Figure 4. ViSOR supports multiple video formats, search by keywords, by video meta-data (e.g., author, creation date, ...), by camera information and parameters (e.g., camera type, motion, IR, omni-directional, calibration).

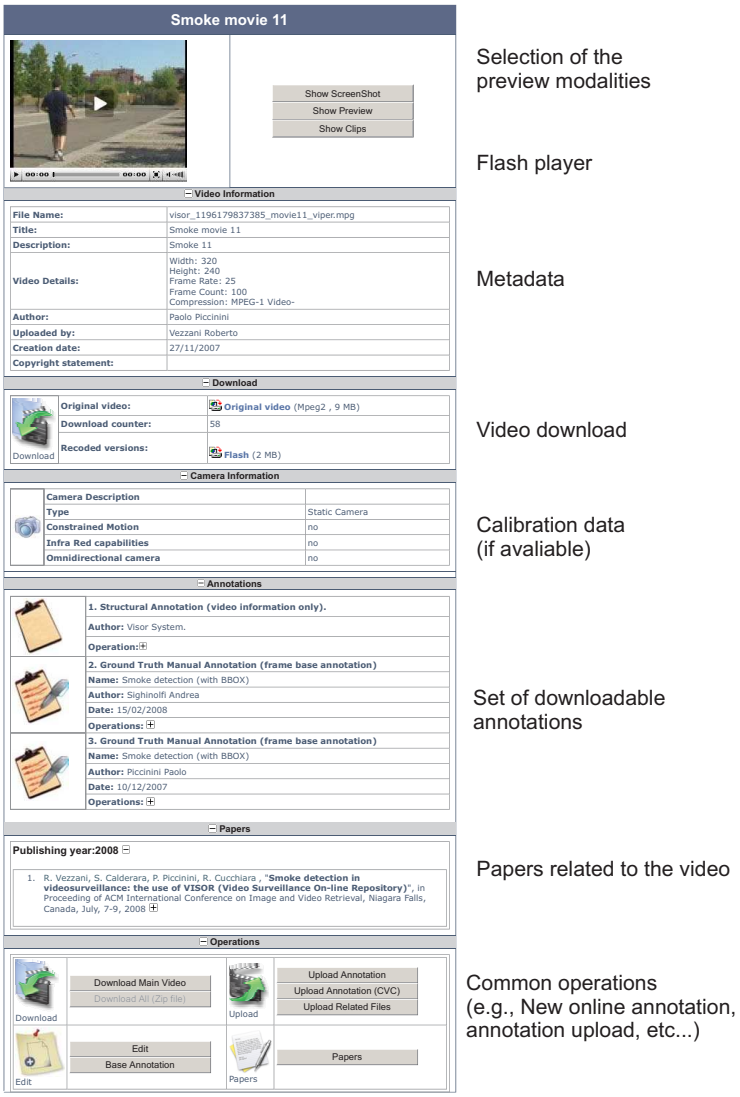


Fig. 4 Video Details

4 ViSOR Video surveillance ontology

"Traditionally listed as a part of the major branch of philosophy known as metaphysics, ontology deals with questions concerning what entities exist or can be said to exist, and how such entities can be grouped, related within a hierarchy, and subdivided according to similarities and differences". Following this definition, our first goal is to collect

entities concerning video surveillance and finding out their possible relations. Actually, video surveillance, as well as most computer vision application fields, has its own set of most significant entities, terms, hierarchies, and relations. Due to the very large set of possible cases combined with the flexibility of natural language, the definition of unique video surveillance ontology is very ambitious and probably unfeasible. Nonetheless, a set of events and entities can be selected due to their importance or their frequent use by the research community.

For example, the multimedia research community has created and usually refers to the 101-concept list of UvA [24] and LSCOM [6] for detecting concepts in videos especially for news and broadcast TV. Most of these concepts have been defined by researchers for the purpose of evaluating automatic techniques for video classification. Similarly, we are proposing a video surveillance ontology, starting from these well known ontologies. Since UvA and LSCOM lists have been defined for generic contexts, only a subset of concepts have been elicited for video surveillance. In addition, UvA and LSCOM concepts are key-frame based only and are not sufficient to describe activities or events. An extension of the original LSCOM list has been taken into consideration (LSCOM Revised Event/Activity Annotations: video-based re-labeling of 24 LSCOM concepts [25]), but only few concepts refer to surveillance.

The surveillance community has made some proposals for event detection ontologies. An example is ontology as defined in the Etiseo project [12] or the result of the "Challenge Project on Video Event Taxonomy" sponsored by the Advanced Research and Development Activity (ARDA) [26]. In [27], a Video Event Representation Language (VERL) is presented which describes an event ontology, associated with Video Event Markup Language (VEML) for event instance annotation.

In order to make this ontology simple to use and flexible at the same time, we created a simple *concept list* first, where concepts are hierarchically structured and defined. The concept list can be dynamically enriched by users under the supervision of the ViSOR moderator to ensure homogeneity, uniqueness and to prevent an uncontrolled explosion of the ontology size. At least, the basic "HAS-A" and "IS-A" relations are required when a new concept is added to the ontology, but other relations can be added or inferred by a reasoning system, such as the one used by Bertini *et al* for sport video annotation [28]. For example, temporal information of the concepts can be used in order to infer relation such as "before", "after", "contextually", and so on. Other details are reported in Section 6, which describes the OWL annotation format.

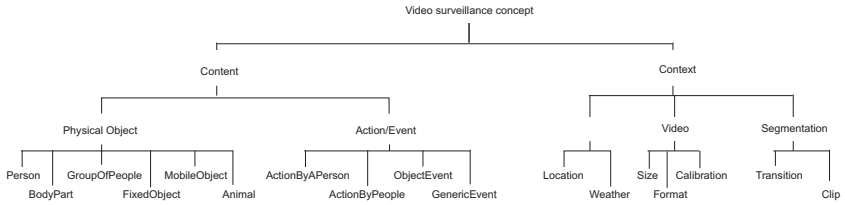


Fig. 5 The Hierarchical taxonomy of ViSOR ontology

The current concept list allows us to classify video shapes, objects and highlights meaningful in a surveillance environment. As depicted in Fig. 5, a generic *concept* can describe either the *context* of the video (e.g., indoor, traffic surveillance, sunny day), or the *content* (e.g., a building, a person, a car). In addition, the *content* can be either a *physical object* which characterizes or appears in the scene (e.g., a building, person, animal), or an *action/event* (e.g., falls, explosion, a interaction among people).

We have collected the most interesting concepts for video surveillance applications and grouped them into 17 main classes relating to *context*, *physical object* and *action/events*, as reported in at bottom level of the diagram in Fig. 5.

A video annotation can be thus considered as a set of *instances* of these classes; a list of related concepts has been assigned to each instance. Some of them directly describe the instance nature, i.e., they are connected to the entity with a "IS-A" relation (e.g., concepts like man, woman, baby, terrorist can be a specialization of the *person* class and can be consequently used to describe instances of that class). Other concepts, instead, describe some instance characteristics or properties, in a "HAS-A" relation with it (e.g., the contour, color, position, bounding box can be descriptive features of *FixedObject* instances). For example, the annotation of a video containing a child riding a bicycle may be composed by an instance of the *Person* class and an instance of the *MovingObject* class. The first item can be further described by the "IS-A" concepts *male*, *child*, *civilian* and so on, while the second item can be specialized with the *Bike* concept. Moreover, if available, the frame by frame centroid position or the bounding box of both items can be provided, using "HAS-A" spatial descriptors.

The *Action/Event* classes are used to annotate concepts like "Explosion", "Person Enter a Scene", "Abandoned Object", "Car Accident", and so on. In video surveillance, the event-based annotation is actually very widespread and using the ViSOR ontology is possible to create content-based annotations (commonly used in the multimedia field) as well as event based annotations (specially conceived for surveillance applications).

Actually, a preliminary list of about 300 surveillance concepts has been defined and can be directly downloaded from the ViSOR web interface [2].

5 Annotation types for surveillance videos

We have already defined video annotation as a set of instances of surveillance classes described with a list of related textual concepts. Temporal and spatial information

about these instances can be provided, depending on the particular application or desired completeness of the annotation. Furthermore, only occasionally an annotation takes into account all the ontology concepts. As a matter of fact, different types of annotation can be generated depending on the drill-down depth used to annotate the video. The choice of the annotation type and detail level is strictly dependent on the application goal; moreover, two annotations can be compared only if they share the same structure and the same detail level. In [29] Kasturi *et al* also defined some rules to evaluate the quality of the annotation; even in this case, it is necessary to define the goal of the annotation to measure how a particular annotation instance meets the desired requirements. To this end, we defined three directions based on which an annotation can be differently detailed: the *temporal level*, the *spatial level*, and the *domain level*.

Intrinsically and from their nature, the defined concepts can be differently related with the time space, depending on the time interval during which the object is visible or the event/action is occurring. For example, some concepts can be associated to the *whole video* (e.g.: *indoor*, *outdoor*), others to a *clip/temporal interval* (e.g., person in the scene) and others to a *single frame/instant* (e.g., *explosion*, *person entering the scene*). But, even if an object is visible in a temporal fragment of the video only, the corresponding annotation can specify the presence of the concept without providing temporal details. Thus, it is very important to define the detail level adopted during the annotation process, otherwise misunderstandings can be generated assuming a different temporal drill-down depth. To this end, we have defined three *temporal* description levels:

– *none or video-level*: no temporal information is given;

- *clip*: the video is partitioned into clips and each of them is described by the set of descriptor instances;
- *frame*: the annotation is provided frame by frame.

Suppose we have an indoor video V_1, in which a child is sitting on a chair. Using the previously defined ontology, we can annotate the video with four descriptor instances $I = \{I_1, I_2, I_3, I_4\}$. I_1, for type Location, is used to annotate the video type and therefore the instance is detailed with the *Indoor* concept. The other concepts are *Person*, *StaticObject*, and *ActionByAPerson*, which can be used to describe the *child*, the *chair* and the "sitting action". A Video Level Annotation does not contain any temporal information, but this does not mean that the child is visible all the time; a Video-Level Annotation indicates only which objects appear and which events occur during the video, nothing else. This can be used for the retrieval engine to access the video repository and search, by way of example, a video containing a child. A Clip level Annotation requires instead a partitioning of the Video into Clips defined by temporal boundaries. An instance I_n relates to a clip, but, again, persistence of the instance during all the clip is not required. This is important in order to select only the part of the video where the action occurs, avoiding to download the entire video. Finally, a Frame-level annotation specifies frame-level temporal boundaries for each entity. If temporal information of an instance is missing, the instance should be visible during all the video.

Similar considerations can be made for the spatial level. Using the previous example, we can specify *where* the child is in the scene in terms of position (e.g., the centroid coordinates), region (e.g., the bounding box) or by giving the complete pixel-level mask. In brief, we can have the following four *spatial* levels:

- *none (image level)*: no spatial information is given and the concept refers to the whole frame;
- *position*: the location of the concepts is specified by an individual point, e.g. the centroid;
- *ROI*: the region of the frame containing the concept is reported, for example using the bounding box;
- *mask*: a pixel level mask is reported for each concept instance.

Considering the domain level, an annotation can be made by taking into account all the ViSOR concepts, in order to provide a description of the video content as detailed as possible. Conversely, in some cases only a few concepts are considered; for example, in a people tracking system only a subset of the person class concepts is used. Some applications are specifically dedicated to a single concept detection (e.g., smoke detection) and include details on that particular concept only. Finally, each video can be described by a set of metadata (such as file name, frame rate, frame size and so on) related to the file itself and not to the semantic content of the video. In this case, we are talking of syntactical annotation only in opposition to the semantic annotation previously defined. Specifying the type of annotation by means of a conceptual level is important in order to infer if the lack of a concept in an annotation implies its real absence in the video or not. To summarize, we have defined the four *conceptual* levels outlined below:

- *none (Syntactical level)*: no semantic information is provided; free-text keywords and title can be provided.
- *one concept*: only one particular concept is considered and annotated; other concepts can be added but they are not the focus of the annotation itself;

Fig. 6 Sample frames for the three presented case studies: smoke detection (top), people tracking with occlusions (middle), and action recognition (bottom)

- *subset*: only a subset of the ViSOR surveillance concepts is considered and the subset adopted should be indicated;
- *whole ontology*: all the ViSOR surveillance concepts are considered.

To provide a better understanding of this annotation taxonomy and, at the same time, give an idea of the videos and concepts that ViSOR includes, we have reported three case studies related to common video surveillance problems. For each of them, we have reported some sample frames from the ViSOR dataset (Fig. 6).

5.1 Case study 1 - smoke detector

A smoke detector or smoke alarm is a device that detects smoke and triggers an alarm to alert people nearby that there is a potential fire. A computer-vision smoke detector should perform the same task, i.e. identify the presence of smoke generating an alarm. In addition to detection, a computer vision system can enrich the knowledge with visual smoke characteristics, for example, with its position and its *size* (i.e., smoke diffusion in the scene). In addition, a video surveillance system can provide information about people interacting with the said smoke or people that have been around the area where

the smoke comes from. The ViSOR concept list contains the following concepts, which are particularly significant in relation to the smoke detection problem:

- the *smoke* concept, which is considered as an "IS-A" attribute of the mobile object class;
- the geometrical features of the smoke, like *position*, *contour* and *bounding box*;
- the *person concepts*, which can be used to describe people interacting with the smoke or located in the place it comes from (both 'IS-A" and "HAS-A" attributes).

A smoke annotation should at least specify the *clip* inside the video when smoke is present and the *region* where it appears; since other details are optional, the *smoke* concept only is interesting. Thus, we can specify the requirements for a typical smoke annotation in terms of the three previously defined levels:

$$SmokeAnnotation \rightsquigarrow \begin{cases} spatial \geq position \\ temporal \geq clip \\ conceptual \geq oneconcept \end{cases} \quad (1)$$

5.2 Case study 2 - tracking systems

Traditionally, a tracking system for video surveillance applications is integrated in a more complex framework that performs several tasks, such as moving object detection, object classification, object localization and feature extraction. Sometimes, it detects and recognizes people and objects interactions. Consequently, taking into account these considerations we have enriched the above mentioned concept list with the following elements.

- "IS-A" concepts related to people, like *man*, *woman*, *child*, *Group of People*, and so on;
- "IS-A" concepts related to interesting objects, both moving and fixed ones, like *vehicle*, *tree*, *building*;
- geometrical features of the tracked object, such as its *position*, *contour*, and *bounding box*;
- event/action that can be used to describe important situations, like *Person Enters A Scene*, *People Aggregation*, *Person falls down* and so on (both 'IS-A" and "HAS-A" attributes).

A typical tracking annotation requires the frame temporal level, as well as a mask region for each tracked object/person. The interesting concepts are a subset of the whole ontology, therefore:

$$TrackingAnnotation \rightsquigarrow \begin{cases} spatial \geq RoI \\ temporal = frame \\ conceptual \geq subset \end{cases} \tag{2}$$

5.3 Case study 3 - action recognition

Action recognition is another crucial topic of the surveillance research area. In this case the abstraction level and the specificity degree required can be very different depending on the application. Thus, collecting an exhaustive concept list is unfeasible. Nevertheless, added the terms most widely used in the ViSOR ontology, such as:

- "IS-A" concepts related to people, like *man*, *woman*, *child*, *Group of People*, and so on;
- "IS-A" concepts related to interesting actions, like *walking*, *running*, *waving*;

– geometrical features of the tracked object, such as its *position*, *contour* and *bounding box*.

In particular, we have added the terms used in the work of Calderara *et al* [30]. Fig. 6 (bottom) shows some video frames for action recognition included in ViSOR. A typical video for action recognition in ViSOR contains atomic actions; thus, the corresponding annotations has no temporal details; the action is performed by the actor in the center and the spatial location is not useful. The only important piece of information stored in such annotations is the specific concept selected among the whole ViSOR ontology.

$$ActionAnnotation \rightsquigarrow \begin{cases} spatial \geq none \\ temporal \geq none \\ conceptual = wholeOntology \end{cases} \tag{3}$$

6 Annotation storage and export formats

The ViSOR annotations are stored in a DBMS. An excerpt of the implemented data schema is reported in Fig. 7. Each video and annotation are cataloged together with a set of metadata, such as author, creation date and description. The ontology described in section 4 is also stored in the database by means of three tables: the "Descriptor" table containes the classes of Fig. 5, while the list of IS-A and HAS-A concepts are stored in the homonym tables. Finally, each annotation is composed of a set of records in the *DescriptorInstance* table, and for each descriptor the set of IS-A and HAS-A related concepts populate the *IsAConceptInstance* and *HasAConceptInstance* tables.

Using a DBMS for annotation storage rather than a set of XML files offers different advantages:

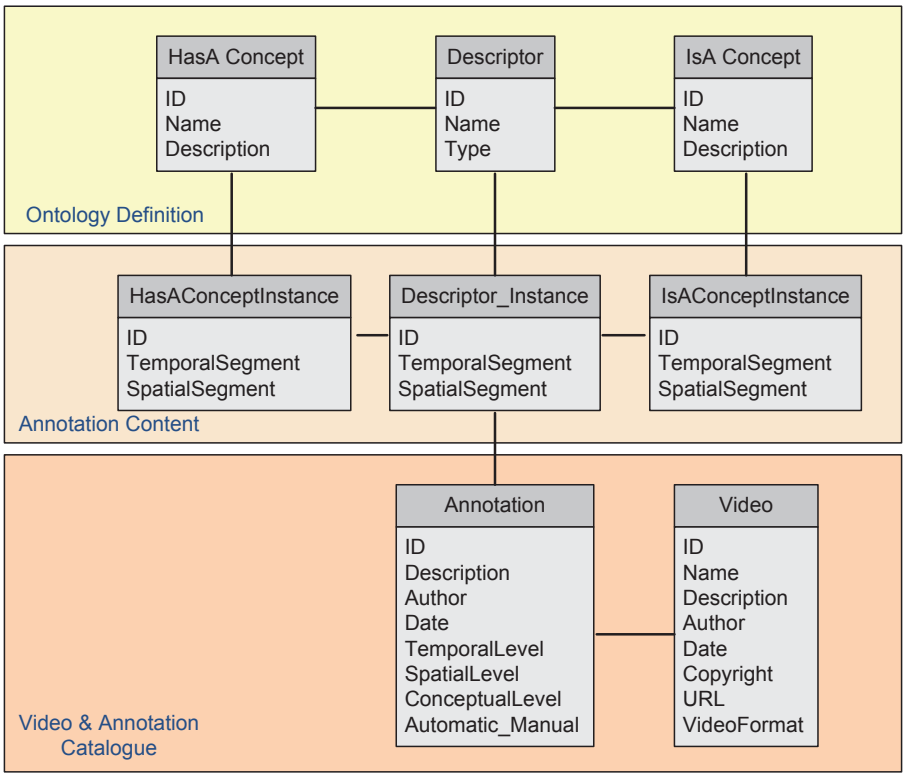


Fig. 7 Excerpt of the database schema used for storing annotations

- multiple formats can be used to export annotations and ontology; currently, three exportation modules (ViPER-XML, OWL, and MPEG7) are available, but it is possible to add new and custom formats
- changes to the reference ontology can be automatically propagated to all the annotations; this way the downloaded annotations are always synchronized with the latest ontology version
- a subset of the annotation items can be downloaded without downloading all the annotation data; the annotation manager can make queries, filters, select the required data
- it is possible to merge data coming from different annotations

– queries on the annotation data are faster and easier.

Currently, the annotation data stored in the DB can be exported in three different annotation formats: ViPER, MPEG7 and OWL. The native annotation format supported by ViSOR is **ViPER XML** [8], developed at the University of Maryland, since it satisfies several requirements: it is flexible, the list of concepts is customizable and it is widespread (e.g., it is used by *Pets* [9] and *Etiseo* [12]). Kasturi *et al* in their very recent work [29] on performance evaluation adopted the ViPER format as well. Differently from other existing tools which are working on textual annotation only, a set of data types which can be used for annotation has been defined. Also, an annotation tool has already been developed by the same authors of the standard (namely, ViPER-GT [7]). Finally, it is possible to achieve a frame level annotation which is more appropriate than the clip level annotation used by other tools.

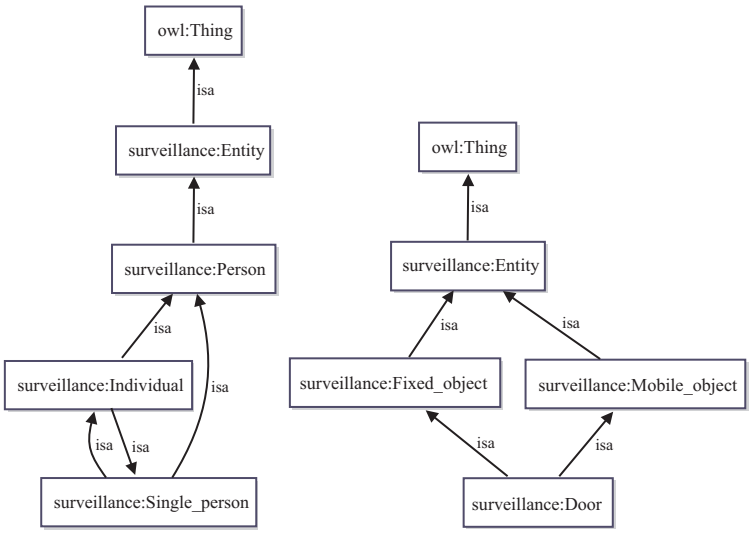
The **MPEG-7** export format follows the requests provided by the VIDI-Video project which is summarized herein. Two sections are defined: the first for ontology definition and the second for annotation data. The ontology part is defined using the *ClassificationScheme* structure. In particular a three level hierarchy of nested Term tags is used. The first level contains three main classes: *Object*, *Action/Event*, and *Context*. Within these classes, the second level defines the classes of Fig. 5; finally the specific concepts are reported in the third level. The annotation part, instead, is based on the *VideoSegment* descriptor. For each *VideoSegment* a *TextAnnotation* section is reported, which contains the concept keyword together with relevance and confidence values, as well as a *MediaTime* section, which indicates the concept temporal interval in the video. *Relevance* set to 1 means that the concept is present, *relevance* set to 0

means that the concept is not present. Also, sometimes the annotator is not sure and may indicate that *confidence* is lower than 1.

6.1 OWL format

The cases presented above and the personal experience of each researcher highlighted that a simple concept list is usually not sufficient to describe and annotate videos. Problems such as duplicates, hierarchies, synonyms and so on cannot be managed with a flat taxonomy of concepts. Neither ViPER nor MPEG7 provide adequate support to a well structured ontology. Web languages such as RDFS go a step further than MPEG7 and ViPER, and could support some constructs, but the Web Ontology Language (OWL) [31] offers a host of other standard properties such as equivalence ("childOf" in an English annotation is the same as "enfantDe" in a French one), or those particular properties which are unique (a social security number is associated with one individual only). The OWL ontology version of the ontology contains a three-levels surveillance taxonomy, as in the other cases. In addition to the HAS-A and IS-A basic relations, semantical improvements have been added to the OWL ontology version. For example, OWL allows us to manage synonyms and multiple inheritance. For example, the *Individual* and *Single person* concepts (which are both included in LSCOM ontology) can be considered synonyms; discarding one of them and keeping only one term can lead to difficulties to import existing annotations, but at the same time it is worth to add a direct link between them in the ontology definition (Fig. 8(a)). The second example is related to the *door* concept. Depending on the scene and the application, a door can be a fixed object in the background or a mobile object which dynamically changes its properties during the video. Thus, the *door* concept does not

match the three partition levels defined above. This problem can be managed using the OWL multiple inheritance support (Fig. 8(b)).



(a) synonyms (b) multiple inheritance

Fig. 8 Examples of synonyms and multiple inheritance concepts in the ViSOR ontology

7 Performance Evaluation Desk

Performance evaluation is still a key task for research communities working on surveillance. Performance evaluation techniques are needed, of course, to measure the progress of research in this area and to compare, for example, different tracking methods. However, there is another and equally important reason for creating evaluation metrics and techniques. In the course of research on a tracking method, there is the need to compare different versions, approaches, or even results of different control parameter settings. The Etiseo [12] and Argos [32] projects have made some important progress in this direction, proposing new metrics, sharing some videos, and calling for a shared evaluation procedure. Anyway, a complete and automatic evaluation desk is not available

yet. With automated, quantitative evaluation techniques, system results coming from different versions or different settings are formally compared. Performance evaluation is thus very important in the context of people tracking, as it is not easy to obtain shared videos and the corresponding reference data for tracking i.e., the ground-truth. The ViSOR framework, instead, makes both videos and ground-truth annotations freely available.

Tests of video surveillance systems can be performed in different ways; by way of example, the background subtraction, the shadow detector and the tracking tasks may need specific tests. Thus, ViSOR has been designed to be flexible and extensible enough to cope with different levels of annotation. Details on two different evaluation types are given in the following.

7.1 Performance Evaluation of tracking systems

People tracking is a common task in surveillance; we have therefore integrated an existing tracking performance evaluation system in ViSOR, namely ViPER-PE [8]. The descriptor to be considered for evaluation (i.e., the *person* descriptor), distance measurement, tolerance thresholds and some filters are specified in a configuration file which has been described in [33] and provides a frame by frame comparison of the bounding box of detected people, reporting both metrics on the detection and the localization of the targets. In particular, ViSOR integrates a people detector and tracker, namely SAKBOT [34] which generates automatic annotations. These annotations can be compared with a manually provided ground truth. An excerpt of the ViSOR output obtained with the described schema is reported in Table 2; in particular, the mean precision and recall -both at object and bounding box levels-

over the whole video are reported. In addition, a frame by frame evaluation can be performed in order to have multiple precision and recall values. For details on the comparison results, refer to the ViPER-PE documentation [8].

Table 2 Some output metrics of the Performance Evaluation for Tracking systems

Total for all frames	
Detection Accuracy:	0.962
Object Count Recall:	1
Object Count Precision:	0.928
Bounding Box measures	
Pixels matched:	545274.0
Pixels missed:	185624.0
Pixels falsely detected:	152948.0
Localized Object Area Recall:	0.896
Localized Box Area Precision:	0.64

7.2 Evaluation of Concept detection

A prerogative of automatic surveillance systems is the high level event detection, ranging from the detection of people entrance in a scene to the detection of people actions, gestures and so on. At this conceptual level, the performance evaluation can be performed in terms of concept retrieved, using metrics and schema coming from the multimedia community (see for example the TRECVid competition [5]). The annotations should have a *conceptual* level as high as possible (*whole ontology* is desirable), but no temporal or spatial information are required. The *Concept detection (Precision and Recall)* evaluation embedded in ViSOR, for example, requires two annotations with the same conceptual level (both whole ontology or at least with the same subset of concepts) and computes $Precision$, $Recall$, and $F - Measure$ [35] values. Fig 9 shows the ViSOR output obtained comparing two different annotations (a manual and an automatic one) provided for the same video.

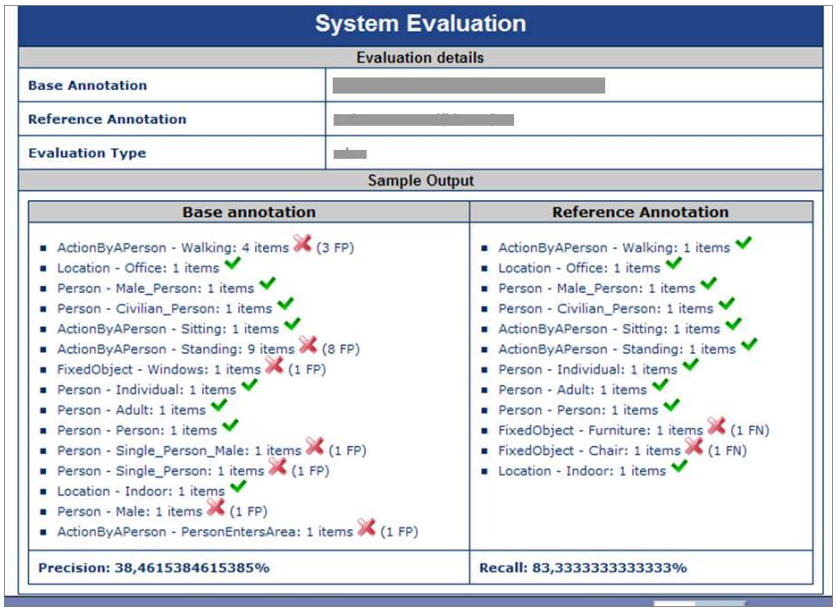


Fig. 9 ViSOR performance evaluation of concept detection; Precision and Recall are obtained matching the concepts included into two compared annotations

8 Conclusion

ViSOR is a dynamic repository for annotated video sequences related to surveillance applications. A suitable ontology for surveillance domains has been defined in order to assure enhanced and easier interoperability among users; furthermore, it can be used in different applications thanks to its flexible structure. In addition, a performance evaluation environment based on the ViPER-PE tool has been integrated in the system and utilized by ViSOR users to evaluate their own systems.

Currently, ViSOR contains 184 videos grouped into 15 categories, as reported in Table 8. A screenshot of the web category selection is reported in Fig. 10.

This project has been recently started and, even if the interface and the database structure have been developed, the population of the database is just at an initial stage. Nonetheless, its interactive interface and the freely available tool set are key points to

Fig. 10 ViSOR video categories

become a reference repository of surveillance and security videos for many multimedia applications.

9 Acknowledgments

This work is supported by the project VIDI-Video (Interactive semantic video search with a large thesaurus of machine-learned audio-visual concepts), funded by E.C. FP6.

References

1. "Vidi-video web site," Website, 2007, http://www.vidivideo.info.
2. "Visor web site," Website, 2007, http://www.openvisor.org.
3. R. Vezzani and R. Cucchiara, "Visor: Video surveillance on-line repository for annotation retrieval," in *ICME*, Hannover, Jun. 2008.

Table 3 Summary of the videos uploaded in ViSOR

Category Name	Description	#Videos	Total Frames
Unimore Outdoor Multicamera	Outdoor Unimore D.I.I. setup - Multicamera	33	400.175
Multicamera - disjoint views	Outdoor Unimore D.I.I. setup - disjoint cameras	16	169042
Door surveillance	Video surveillance of entrance doors	6	470.912
Human actions	Videos of different human actions	40	10396
Human Actions II	Long videos for human action recognition	1	4.182
Indoor Domotic	Indoor Domotic Unimore D.I.I. setup	16	17.727
Indoor People Tracking	Video for indoor people tracking with occlusions	6	3611
IseLab Collection	Videos from the IseLab - Computer Vision Center - Universitat Autnoma de Barcelona - Road crossing of pedestrians and vehicles	3	4.960
Other	Other	6	2498
Outdoor Unimore	Outdoor Unimore D.I.I. setup - Single Camera	27	77.211
Outdoor Face	Outdoor Video For Face detection	3	2.603
Shadows	Shadows	5	3.306
Smoke	Videos for Smoke detection	14	25.570
Stopped Vehicles	Videos for Stopped Vehicles Detection	4	12.168
VSSN06 Competition	Videos used for the VSSN background competition	4	11.644
	TOTAL	184	1.216.005

4. P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The feret evaluation methodology for face-recognition algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090–1104, 2000.
5. A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation campaigns and trecvid," in *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*. New York, NY, USA: ACM Press, 2006, pp. 321–330.
6. M. Naphade, L. Kennedy, J. R. Kender, S.-F. Chang, S. J. R., P. Over, and A. Hauptmann, "A light scale concept ontology for multimedia understanding for trecvid 2005," IBM Research, Tech. Rep., 2005.
7. "Viper toolkit at sourceforge," Website, 2005, http://viper-toolkit.sourceforge.net/.

8. D. Doermann and D. Mihalcik, "Tools and techniques for video performance evaluation," *Proc. of Int'l Conference on Pattern Recognition*, vol. 04, p. 4167, 2000.
9. "Pets: Performance evaluation of tracking and surveillance," Website, 2000–2007, http://www.cvg.cs.rdg.ac.uk/slides/pets.html.
10. *VSSN '06: Proceedings of the 4th ACM international workshop on Video surveillance and sensor networks*. New York, NY, USA: ACM, 2006, general Chair-Jake K. Aggarwal and General Chair-Rita Cucchiara and Program Chair-Andrea Prati.
11. H. O. S. D. Branch, "i-lids - imagery library for intelligent detection systems," Website, 2006, http://scienceandresearch.homeoffice.gov.uk/hosdb/.
12. A.-T. Nghiem, F. Bremond, M. Thonnat, and V. Valentin, "Etiseo, performance evaluation for video surveillance systems," in *Proceedings of AVSS 2007*, 2007.
13. "Behave," Website, http://homepages.inf.ed.ac.uk/rbf/BEHAVE/.
14. "Candela," Website, http://www.extra.research.philips.com/euprojects/candela/.
15. "Caviar," Website, http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/.
16. "Objectvideo virtual video," Website, http://development.objectvideo.com/.
17. C. Machy, X. Desurmont, J.-F. Delaigle, and A. Bastide, "Introduction of cctv at level crossings with automatic detection of potentially dangerous situations," in *2nd Selcat Workshop*, 2007.
18. "Surveillance performance evaluation initiative (spevi)," Website, http://www.spevi.org.
19. "Image sequence server of the institut fr algorithmen und kognitive systeme," Website, http://i21www.ira.uka.de/image_sequences/.
20. D. Weinland, R. Ronfard, and E. Boyer, "Free viewpoint action recognition using motion history volumes," *Comput. Vis. Image Underst.*, vol. 104, no. 2, pp. 249–257, 2006.
21. "Trecvid 2008 surveillance video," Website, http://www-nlpir.nist.gov/projects/tv2008.
22. "Cmu graphics lab motion capture database," Website, http://mocap.cs.cmu.edu/.
23. "Humaneva - synchronized video and motion capture dataset for evaluation of articulated human motion," Website, http://vision.cs.brown.edu/humaneva/.
24. C. Snoek, M. Worring, J. Van Gemert, J. Geusebroek, and A. Smeulders, "The challenge problem for automated detection of 101 semantic concepts in multimedia," in *Proceedings of the 14th ACM Int'l Conference on Multimedia*. New York, NY, USA: ACM, 2006, pp. 421–430.

25. L. Kennedy, "Revision of lscom event/activity annotations, dto challenge workshop on large scale concept ontology for multimedia," Columbia University ADVENT, Tech. Rep., 2006.
26. R. Nevatia, J. Hobbs, and B. Bolles, "An ontology for video event representation," in *CVPRW '04: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04) Volume 7*. Washington, DC, USA: IEEE Computer Society, 2004, p. 119.
27. A. R. Francois, R. Nevatia, J. Hobbs, and R. C. Bolles, "Verl: An ontology framework for representing and annotating video events," *IEEE MultiMedia*, vol. 12, no. 4, pp. 76–86, 2005.
28. M. Bertini, A. Del Bimbo, C. Torniai, C. Grana, R. Vezzani, and R. Cucchiara, "Sports video annotation using enhanced hsv histograms in multimedia ontologies," in *International Workshop on Visual and Multimedia Digital Libraries*, Modena, Italy, Sep. 2007, pp. 160–167.
29. R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M. Boonstra, V. Korzhova, and J. Zhang, "Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 319–336, 2009.
30. S. Calderara, R. Cucchiara, and A. Prati, "Action signature: a novel holistic representation for action recognition," in *5th IEEE International Conference On Advanced Video and Signal Based Surveillance (AVSS2008)*, Sep. 2008.
31. F. van Harmelen, J. Hendler, I. Horrocks, D. McGuinness, P. F. Patel-Schneider, and L. A. Stein, "Owl web ontology language reference," http://www.w3.org/TR/owl-ref/, 2002.
32. P. Joly, J. Benois-Pineau, E. Kijak, and G. Qunot, "The argos campaign: Evaluation of video analysis tools," in *Fifth International Workshop on Content-Based Multimedia Indexing (CBMI'07)*, 2007.
33. R. Vezzani and R. Cucchiara, "Annotation collection and online performance evaluation for video surveillance: the visor project," Santa Fe, New Mexico, Sep. 2008.
34. R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts and shadows in video streams," *IEEE Transactions on Pattern Analysis and Machine*

Intelligence, vol. 25, no. 10, pp. 1337–1342, Oct. 2003.

35. C. J. V. Rijsbergen, *Information Retrieval*. Newton, MA, USA: Butterworth-Heinemann, 1979.