Annotation Collection and Online Performance Evaluation for Video Surveillance: the ViSOR Project

Roberto Vezzani, Rita Cucchiara
Imagelab - Dipartimento di Ingegneria dell'Informazione
University of Modena and Reggio Emilia, Italy
{roberto.vezzani, rita.cucchiara}@unimore.it

Abstract

This paper presents the Visor (VIdeo Surveillance Online Repository) project designed with the aim of establishing an open platform for collecting, annotating, retrieving, sharing surveillance videos, and of evaluating the performance of automatic surveillance systems. The main idea is to exploit the collaborative paradigm spreading in the web community to join together the ontology based annotation and retrieval concepts and the requirements of the computer vision and video surveillance communities. The ViSOR open repository is based on a reference ontology which integrates many concepts, also coming from LSCOM and MediaMill ontologies. The web interface allows video browse, query by annotated concepts or by keywords, compressed video preview, media download and upload. The repository contains metadata annotations, which can be either manually created as ground truth or automatically generated by video surveillance systems. Their automatic annotations can be compared each other or with the reference ground-truth exploiting an integrated on-line performance evaluator.

1. Introduction

All the researchers working in video surveillance and computer vision suffer of the lack of common video database, annotation and ground truth data to provide fair performance evaluation and open discussions about techniques and methodologies. In other research contexts large databases are available such as FERET for biometry or TRECVID for video retrieval and corresponding ontologies have been defined such as the LSCOM concept list. In video surveillance many projects in performance evaluation are growing their popularity with some performance evaluation tools and available dataset. Some open source tools, such as ViPER-GT and ViPER-PE [11] constitute an interoperable platform to manually select concepts and events in video, generate ground truth and annotate videos into XML files. The ViPER annotation format [2] is widely exploited by available databases of videos which are created in contexts of workshops and conferences like the PETS workshop series[13] or the VSSN workshops of the ACM Multimedia Conference [12] and in the ones that become available from some European or national projects such as I-Lids[1] and Etiseo [8]. Some examples of available datasets are reported in the table of Figure 3. Most of these datasets have two main drawbacks. The first is their narrow focus on few specific problems of computer vision and pattern recognition. The PETS datasets, for instance, have been deeply exploited in some applications but they have been proposed within their a-priori annotation with the aim of coping a single or few video surveillance problems. The second limitation is the lack of user interaction; for example, user cannot share their own annotation data, or grow the dataset with other videos, or comment them, and so on. Moreover, the defined ontology is normally not available, and there are not graphical tools or querying systems to select only the subset of videos useful for a given application.

The Video Surveillance Online Repository (ViSOR) for annotation retrieval has been conceived to meet these needs. It has been designed and is under development in the context of the Vidi-Video European project of the VI FP. First aim of ViSOR is to gather and make freely available surveillance video footages for the research community on pattern recognition and multimedia retrieval. At the same time, our goal is to create an open forum and a interactive repository to exchange, compare and discuss results of many problems in video surveillance and retrieval.

Together with the videos, ViSOR defines an ontology for metadata annotations, both manually provided as ground truth and automatically obtained by video surveillance systems. Annotation refers to a large ontology of concepts on surveillance and security related objects and events, defined including concepts from LSCOM and MediaMill ontologies. Moreover, ViSOR provides tools for enriching the ontology, annotating new videos, searching by textual queries, composing and downloading videos.

In this paper we describe the ViSOR project, the video surveillance ontology and the web interface for analyzing and querying the results of automatic event detection systems. Finally, the online performance evaluation framework integrated in the system is illustrated.

2. The ViSOR framework

As above mentioned, first aim of ViSOR is to collect and share surveillance video together with metadata annotations. A conceptual schema of the ViSOR framework is depicted in Figure 1.

The system has been conceived as a web application; thus, two main sections can be defined, one working at server side and the other one constituted by the client user interface. The web interface will be presented in Section 4. The core of the system, instead, is the server repository which is composed by three different entities: the *Video storage*, the *annotation database* and the *reference ontology* (see Figure 2). The video storage subsystem contains not only the original uploaded videos, but also recoded versions of them (e.g., an MPEG1 version and a flash compressed preview version), associated textual keywords, relations between videos (e.g., it is possible to specify that two or more videos are different but synchronized views of the same place), and clip segmentation, if any. The ontology subsystem, instead, stores the reference ontology (a video surveillance ontology in our case) using the hierarchical schema fully described in the next section. This ontology can be modified only by the ViSOR manager in order to assure homogeneity and the uniqueness. Finally, for each video one or more annotations can be provided, meaning with annotation a set of instances of the descriptor and the concepts defined in the reference ontology.

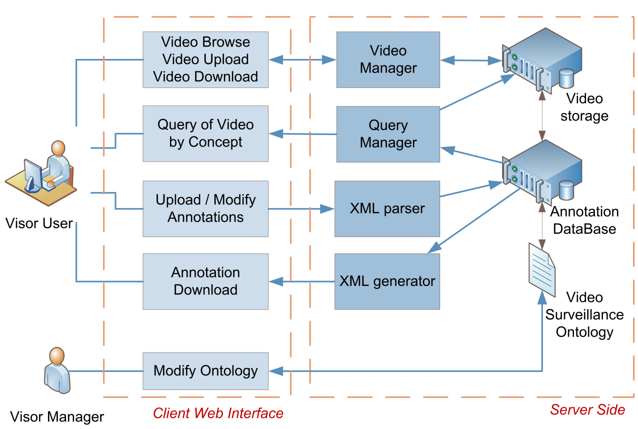

Figure 1. Conceptual schema of the ViSOR framework

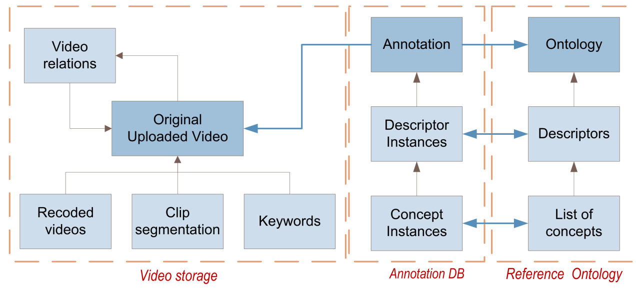

Figure 2. Structural schema of the ViSOR core

3. Video Surveillance Ontology in ViSOR

Some proposals of ontologies for event detection in surveillance have been carried out. An example is the ontology defined in the Etiseo project [8] or the result of the “Challenge Project on Video Event Taxonomy” sponsored by the Advanced Research and Development Activity (ARDA) [7]. In [3] a Video Event Representation Language (VERL) is presented which describes an event ontology, associated with Video Event Markup Language (VEML) for event instance annotation. ViPER-GT [11], instead, is a very spread graphical tool for manual annotation of objects and frame-based events, exploited in video-surveillance community.

Here we start from the ViPER framework and propose an open simple ontology structured as a simple “concept list”: this taxonomy is a basic form of ontology where concepts are hierarchically structured and univocally defined. The concept list can be dynamically enriched by users under the supervision of the ViSOR moderator to ensure the homogeneity and the uniqueness. The goal is to create a very large concept list avoiding synonymy and polysemy drawbacks.

3.1. Video Surveillance Concepts

We defined a basic taxonomy to classify video shapes, objects and highlights meaningful in a surveillance environment. A “concept” can describe either the *context* of the video (e.g., indoor, traffic surveillance, sunny day), or the *content* which can be a *physical object* characterizing or present in the scene (e.g., building, person, animal) or a detectable *action/event* occurring (e.g., falls, explosion, interaction between people).

The defined concepts can be differently related with the time space. Thus, we defined a time based taxonomy of the concepts depending on its span, e.g. the time interval during which the object is visible or the event/action is occurring. A concept can be associated to the *whole video* (e.g.: indoor, outdoor), to a *clip/temporal interval* (e.g., person in the scene), or to a *single frame/instant* (e.g., explosion, person entering the scene).

A first reference concept list has been obtained as a sub-

Dataset		Website	Topics	Ground-Truth	Size
BEHAVE		http://homepages.inf.ed.ac.uk/rbf/BEHAVE/	Unusual activities	yes	8 with ground truth
CANDELA		http://www.multitel.be/~va/candela/	Indoor left-luggage and traffic monitoring on road intersection	no	16 indoor
Etiseo		http://www-sop.inria.fr/orion/ETISEO/	Object Detection, Object Localization, Object Tracking, Object Classification.	yes	86 video clips
i-Lids (AVSS 2007)		ftp://motinas.elec.qmul.ac.uk/pub/iLids/	Stopped vehicles and abandonded luggage	yes	14 sequences
ObjectVideo Virtual Video		http://development.objectvideo.com/	Tool to generate virtual video sequences for surveillance purposes.	yes	-
PETS	2001	http://www.cvg.cs.rdg.ac.uk/PETS2001/pets2001-dataset.html	Outdoor people and vehicle tracking	yes	5 sequences
	2002	http://www.cvg.cs.rdg.ac.uk/PETS2002/pets2002-db.html	Indoor people tracking (and counting)	yes	6 sequences
	2004	http://www-prima.inrialpes.fr/PETS04/caviar_data.html	People tracking and activity recognition	yes	28 sequences, 6 scenarios
	2006	http://pets2006.net/	Surveillance of public spaces, detection of left luggages	yes	7 datasets (4 camera views each one)
	2007	http://pets2007.net/	Multisensor sequences containing loitering, attended luggage removal (theft), and unattended luggage	yes	8 datasets (4 camera views each one)
SELCAT		http://www.multitel.be/~va/selcat/	Level crossing monitoring for stopped vehicles detection.	yes	8 sequences
SPEVI		http://www.spevi.org	Face detection and tracking	partial	10 sequences
Traffic datasets by Institut für Algorithmen und Kognitive Systeme		http://i21www.ira.uka.de/image_sequences/	Traffic surveillance in particular on road intersections	no	14 sequences
VISOR		http://imagelab.ing.unimore.it/visor	Indoor and outdoor surveillance sequences; annotation data for object detection, tracking, events, and much more.	yes	65 sequences at 12/12/2007 (in progress)
VSSN		http://imagelab.ing.unimore.it/vssn06/	background subtraction competition	no	7 sequences

Figure 3. Available surveillance datasets

set of two different predefined sets, respectively the 101-concept list of UvA[9] and LSCOM[6]. Since these lists have been defined for generic contexts, only a subset of the reported concepts have been elicited for video surveillance. Moreover, UvA and LSCOM lists are key-frame based only and are not enough to describe activities and events. An extension of the base LSCOM list have been considered (LSCOM Revised Event/Activity Annotations: video-based re-labeling of 24 LSCOM concepts [5]), but only few concepts refer to surveillance. Thus, we have collected and reported other concepts we are interesting on; most of them are defined at a very high abstraction level. Actually, a preliminary list of more than 100 surveillance concepts has been defined.

The video surveillance concepts can belong to three semantically different categories (*Physical Object, Action/Event, Context*). More precisely, the ViSOR ontology is structured in several classes, each of them belonging to one of the previously defined categories as reported in Table 2. A video annotation can be considered as a set of instances of these classes; for each instance a list of related concepts are assigned. Some of them directly describe the nature of the instance, i.e., they are connected to the entity with a "IS-A" relation (e.g., concepts like man, woman, baby, terrorist can be a sort of specialization of the "person" class and thus they can be use to describe instances of that class). Other concepts, instead, describe some characteristics or properties of the instance, in a "HAS-A" relation with it (e.g., the contour, the color, the position, the bounding box can be descriptive features of *FixedObject* instances).

Specialization relations are always *static*, i.e, they do not change during time; for example, a person can be a man or a woman, but reasonably it cannot switch between them during the video clip. Differently, some "HAS-A" relation can be *dynamic*; for example, the position and the color of the person can be different frame by frame. Thus, we have distinguished the "HAS-A" concepts in *static* and *dynamic*. In other words, the appellation *dynamic* indicates that the concept has a dynamic evolution of some of its visual properties, and thus may be recognized performing an analysis that goes beyond a single key-frame description, or may provide more information if this evolution is taken into account. A complete list of the video surveillance concepts can be directly downloaded from the ViSOR portal.

3.2. Concepts for tracking systems

Traditionally, a tracking system for video surveillance application 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 recognize people and objects in-

Person			
“Is-a” Concepts			
Name	Definition	Type	Dynamic
Adult	Shots showing a person over the age of 18 (LSCOM #181)	True/False	-
Aggressor	(LSCOM #461)	True/False	-
Baby	images of babies (children that are too young to walk) (LSCOM #247)	True/False	-
Boy	One or more male children. (LSCOM #183)	True/False	-
Child	images of children (LSCOM #273)	True/False	-
Civilian_Person	One or more persons not in the armed services or police force. (LSCOM #105)	True/False	-
Female	(LSCOM #21)	True/False	-
Girl	One or more female children. (LSCOM #184)	True/False	-
Male	(LSCOM #17)	True/False	-
Person	Shots depicting a person. The face may be partially visible (LSCOM #217)	True/False	-
Police/security	(LSCOM #42)	True/False	-
“Has-a” Concepts			
Position_BBOX	Bounding box containing the person	rectangle	True
PositionBar	2D Position of the gravity center	point	True
Contour	Person’s Contour	polygon	True
IDPerson	Application defined ID	integer	False
RealHeight	Real height of the person	float	False
PersonName	Person’s Name	string	False
Mobile Object			
“Is-a” Concepts			
Bicycle	A person riding a bicycle. (LSCOM #197)	True/False	-
Bird	(LSCOM #79)	True/False	-
Bus	Shots of a bus (LSCOM #227)	True/False	-
Car	Shots of a car (LSCOM #221)	True/False	-
Chair	(LSCOM #56)	True/False	-
Smoke	Shots with smoke present. (LSCOM #161)	True/False	-
“Has-a” Concepts			
Position_BBOX	Bounding box containing the object	rectangle	True
PositionBar	2D Position of the gravity center	point	True
Contour	Contour of the object	polygon	True

Table 1. Excerpts taken from *Person* and *Mobile Object* concept lists.

teractions. Thus, 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 than fixed ones, like *vehicle*, *tree*, *building*;
- geometrical features of the tracked object, like the *position*, the *contour*, and the *bounding box*;
- event/action that can be used to describe people interacting each other or with objects, like *Person Enters A Scene*, *People Aggregation*, *Person falls down*, and so on (both ‘Is-A” and “Has-A” attributes).

Class	Category
1. Person	PhysicalObject
2. BodyPart	PhysicalObject
3. GroupOfPeople	PhysicalObject
4. FixedObject	PhysicalObject
5. MobileObject	PhysicalObject
6. ActionByAPerson	Action/Event
7. ActionByPeople	Action/Event
8. ObjectEvent	Action/Event
9. GenericEvent	Action/Event
10. Video	Context
11. Clip	Context
12. Location	Context

Table 2. Set of surveillance classes

A list of some surveillance concepts is reported in Table 1 and it has been obtained from the video surveillance concept list of the ViSOR system [10].

3.3. The annotation format

The native annotation format supported by ViSOR is ViPER[2], developed at the University of Maryland. The choice of this annotation format has been made due to several requirements that ViPER satisfies: it is flexible, the list of concepts is customizable; it is widespread avoiding the difficulties to share a new custom format (e.g., it is used by *Pets* and *Etiseo*); it is clear and easy to use, self containing since the description of the annotation data is included together with the data. Differently from other existing tools working only on textual annotation, a set of data types which can be used for annotate has been defined (see

Data Type	Description
bbox	A bounding box; it is a rectangle on the image.
bvalue	A Boolean value: either “true” or “false”.
circle	A circle, in terms of center point and radius.
ellipse	An ellipse, in terms of its bounding box.
fvalue	A floating point number.
lvalue	An enumeration type. In the config part the list of allowed values must be defined.
obox	An oriented bounding box.
point	Some specific pixel in the image.
polygon	A polygon or polyline, given as a list of points.
relation	A set of object identification numbers to a certain type of descriptor.
svalue	A string value. Remember that strings must be xml-escaped.

Table 3. ViPER Data types

Table 3).

Moreover, an annotation tool has already been developed by the same authors of the standard (namely, ViPER-GT [11]). Finally, it is possible to achieve a frame level annotation that is more appropriate than the clip level annotation adopted by other tools.

The annotation data is stored as a set of records. Each record, called *descriptor*, annotates an associated range of frames with a set of attributes. To inform applications of the types of descriptors used to create the data file and the data-types of the associated attributes, users must provide configuration information at the beginning of file. To this aim, ViPER files are basically composed by two sections; the first one is called *config part* and explicitly outlines all possible descriptors in the ViPER file. It defines each descriptor type by name and lists all attributes for each descriptor. From the ViSOR portal a predefined *config file* containing the video surveillance concept list described in the previous section can be obtained. The second section of each ViPER file, namely *data part*, contains instances of the descriptors defined in the *config part*. For each instance, the frame span (i.e., range) of the descriptor visibility together with a list of attributes values are reported. For more details refer to the ViPER manual [11] or directly to the ViSOR portal [10].

4. ViSOR Web Interface

The ViSOR web interface has been designed in order to share the videos and the annotation contents. In Figure 1 the main modules of the ViSOR web interface are shown. In particular, the web interface allows to:

- upload videos, download them exploiting a visual browse interface or a query by keywords;
- retrieve videos by concepts, which mean to look for the desired concept inside the annotation database and return a list of annotations and the related videos containing that concept;
- upload or modify annotation data referred to a video;
- download the annotation of a video;

Since the annotation data is stored in a database, uploading and downloading annotations require an XML parser and a XML generator module respectively.

Figure 4. Some screenshots of the ViSOR web interface

Some screen shots of the web interface are shown 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).

Three modalities have been implemented to allow video access: video preview, based on a compressed stream, single screen shot (a representative frame of the entire video) or a summary view, in which clip level screen shots are reported.

Until now about 80 videos belonging to different scenarios, both indoor and outdoor, have been added to ViSOR but the number of video is growing day by day. Recently, the University of Barcelona has uploaded videos coming from the Hermes project with a pixel-level annotation for tracking evaluation [4]. Videos for shadow detection, for smoke detection, for posture classification, and for background maintenance are stored in ViSOR as well. For example, some widely used videos for shadow detection algorithm evaluation have been added. A screen shot with representatives of this last category is shown in Figure 5 obtained using the Clip level mode.

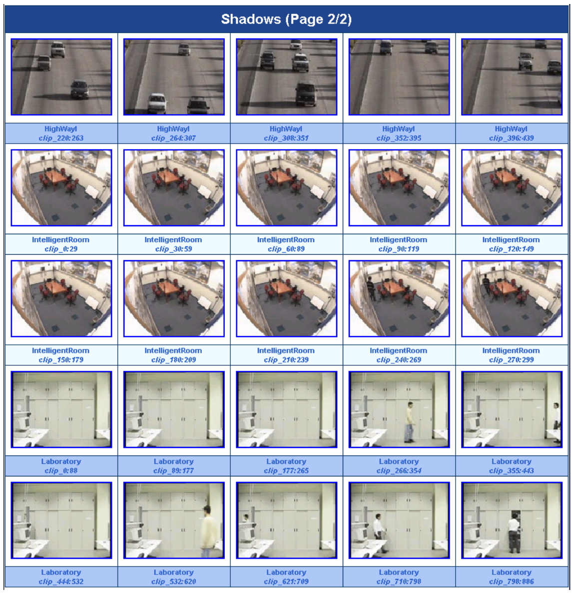

Figure 5. Thumbnails of the ViSOR videos belonging to the shadow detection category using the Clip Mode

Since different annotation levels can be adopt (smoke event detection only, events plus bounding box of the smoke, smoke and people annotation) and both ground truth and automatic annotations can be provided, for each video a set of annotations are shared and available for download. For each annotation, the entire annotation as well as a subset of the annotation fields, filtering by frame number, descriptor or single attribute can be extracted. At the present the annotation can be exported in the ViPER format only, but an MPEG7 format export module is under development.

Moreover, an integrated player for videos and annotations has been done. The player draws geometrical concepts superimposing them to the video; other linguistic concepts are reported below the video. A tree representation of the complete annotation content is reported as well. Finally, a set of descriptor level selection buttons are depicted in order to hide or show the relative annotation data. A screen shot of this player is reported in Figure 6.

5. Performance Evaluation of tracking systems

Performance evaluation is still a key task for research communities working on surveillance. Techniques of performance evaluation are needed, of course, to measure progress of research in this area, and to compare, for example, different tracking methods. However, there is another, 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 settings of control parameters. 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, make freely available both videos and ground-truth annotations. Moreover, we have integrated in the ViSOR framework the performance evaluation tool named ViPER-PE [2].

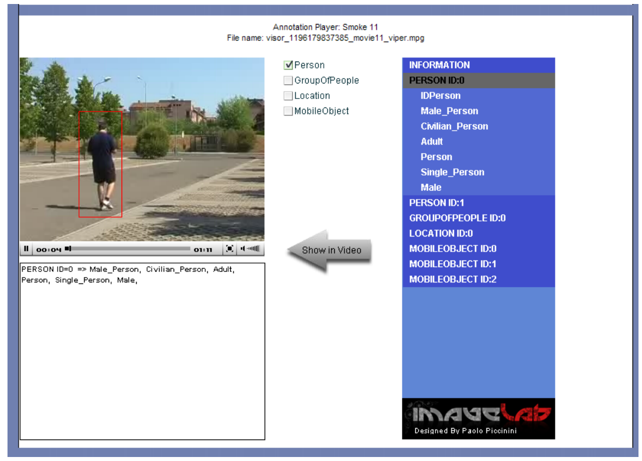

Figure 6. A screen shot of the flash player developed to simultaneously preview video and annotations.

ViPER-PE allows to compare two different annotation files and to report performance results. If the two annotations are coming from system outputs, then the evaluation results can be considered as a system comparison and exploited to choose which system performs well on the selected video. Otherwise, comparing a system output with the video ground-truth, the results will be an objective measure of the system efficacy. In both cases, since both videos, annotation, and performance metrics are the same for each ViSOR users, the performance evaluation is fair and objective.

Figure 7 contains the block diagram of the performance evaluation procedure embedded in ViSOR. Users can upload the annotation generated by their video surveillance systems. These annotations should be written using the XML ViPER format as in Section 3.3 and the reference ontology described in Section 3. At the same time, ground truth annotations can be provided for the same set of videos using the ViPER-GT tool or a similar one. Once more than one annotation is available for the same video, performance

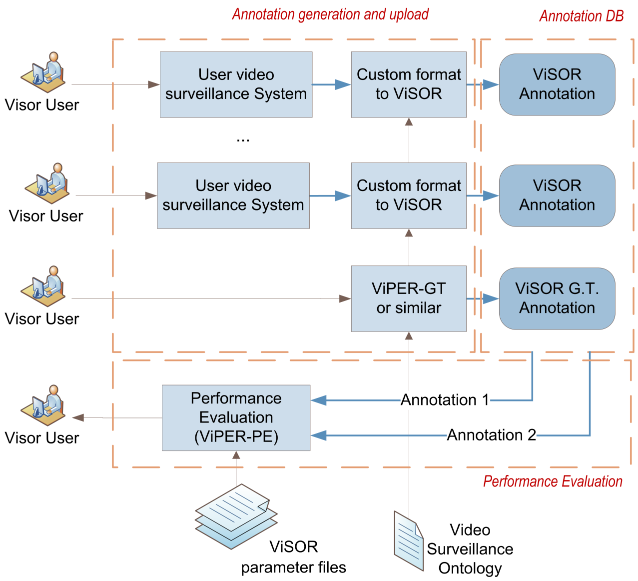

Figure 7. Block diagram of the performance evaluation procedure for a single video using ViSOR

evaluation tests can be carried out selecting two annotations and a performance schema, i.e. a particular ViPER-PE configuration file selected among the set provided by the web interface of ViSOR. The descriptors to be considered for the evaluation (e.g., *person* descriptors), the distance measure, the tolerance thresholds, and some filters are specified in these configuration file.

For example, the evaluation schema for people tracking included in ViSOR takes into account only the *Person* descriptors and compares frame by frame the bounding box of them, reporting both metrics on the detection and the localization of the targets. A screen-shot of the ViSOR output obtained with the described schema is reported in Figure 8 while the correspondent configuration files are shown in Figure 9.

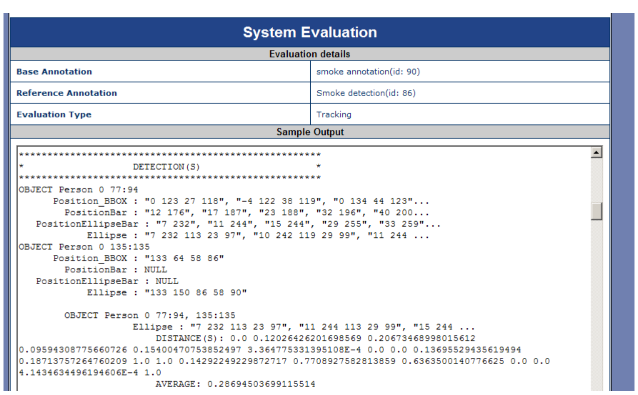

Figure 8. A screen shot of the evaluation system included in ViSOR

6. Conclusion and future work

ViSOR is a dynamic repository of annotated video sequences related to surveillance applications. A suitable ontology for surveillance domains has been defined in order to assure a better and easier interoperability among users. The flexible structure and implementation of the system allows the exploitation on different application. Moreover, a performance evaluation environment based on the ViPER-PE tool has been integrated in the system and used by the ViSOR users in order to evaluate their own systems.

This project (funded by VidiVideo EU project) is recently started and even if the interface and the database structure have been developed, the population of the database is just on an initial stage. Nonetheless, its interactive interface and the free available tool set are key points to become a reference repository of surveillance and security videos for many multimedia applications.

7. Acknowledgments

This work is supported by the project VidiVideo (Interactive semantic video search with a large thesaurus of machine-learned audio-visual concepts), funded by E.C. VI FP. The authors are also thankful to the CVC laboratory of the Autonomous University of Barcelona for providing them videos and annotations.

References

[1] H. O. S. D. Branch. i-lids - imagery library for intelligent detection systems. Website, 2006. http://scienceandresearch.homeoffice.gov.uk/hosdb/.

[2] D. Doermann and D. Mihalcik. Tools and techniques for video performance evaluation. *Proc. of Int'l Conference on Pattern Recognition*, 04:4167, 2000.

[3] A. R. Francois, R. Nevatia, J. Hobbs, and R. C. Bolles. Verl: An ontology framework for representing and annotating video events. *IEEE MultiMedia*, 12(4):76–86, 2005.

[4] J. Gonzlez, F. X. Roca, and J. J. Villanueva. Hermes: A research project on human sequence evaluation. In *Computational Vision and Medical Image Processing (VipIMAGE'2007)*, Porto, Portugal, Oct. 2007.

[5] L. Kennedy. Revision of lscom event/activity annotations, dto challenge workshop on large scale concept ontology for multimedia. Technical report, Columbia University ADVENT, 2006.

[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. Technical report, IBM Research, 2005.

[7] 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*, page 119, Washington, DC, USA, 2004. IEEE Computer Society.

- Evaluation Parameters File (epf file) -

```
#BEGIN_OBJECT_EVALUATION
OBJECT Person [- -]
   Position_BBOX : [dice -]
   Ellipse : [- -]
#END_OBJECT_EVALUATION

#BEGIN_FRAMEWISE_EVALUATION
OBJECT Person
Position_BBOX : dice overlap matchedpixels missedpixels \
falsepixels [arearecall 0.6] [areaprecision 0.7]
Ellipse : dice overlap matchedpixels missedpixels \
falsepixels [arearecall 0.6] [areaprecision 0.7]
#END_FRAMEWISE_EVALUATION
```

- Properties file (pr file)-

```
# Level of analysis
# 3=statistical comparison
level = 3
target_match = MULTIPLE
# Range Distance Metric
# dice = Dice coefficient
range_metric = dice
# String Distance Metric
# L = Levenshtein (Edit distance)
string_metric = L
# Level Specific Metrics
level3_metric = mean
###############################################################
# Default Tolerance Configuration [0 = exact match]
###############################################################
# Temporal Range
range_tol = 0.2
# Attributes, <attribute type>_tol
bbox_tol = 0.25
ellipse_tol = 0.99
# Level Specific
level3_tol = 0.3
###############################################################
# Presentation Parameters
###############################################################
verbose = true
attrib_width = 50
```

Figure 9. Sample configuration files for the tracking evaluation system of ViSOR

[8] A.-T. Nghiem, F. Bremond, M. Thonnat, and V. Valentin. Etiseo, performance evaluation for video surveillance systems. In *Proceedings of AVSS 2007*, 2007.

[9] 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*, pages 421–430, New York, NY, USA, 2006. ACM.

[10] Visor portal. Website, 2007. http://imagelab.ing.unimore.it/visor.

[11] Viper toolkit at sourceforge. Website, 2005. http://viper-toolkit.sourceforge.net/.

[12] *VSSN '06: Proceedings of the 4th ACM international workshop on Video surveillance and sensor networks*, New York, NY, USA, 2006. ACM. General Chair-Jake K. Aggarwal and General Chair-Rita Cucchiara and Program Chair-Andrea Prati.

[13] Pets: Performance evaluation of tracking and surveillance. Website, 2000–2007. http://www.cvg.cs.rdg.ac.uk/slides/pets.html.