An overview of SUNAR

Petr Chmelar, Ales Lanik and Jozef Mlich

Brno University of Technology, Faculty of Information Technology,
Bozetechova 2, 612 66 Brno, Czech Republic
{chmelarp,ilanik,imlich}@fit.vutbr.cz
http://www.fit.vutbr.cz

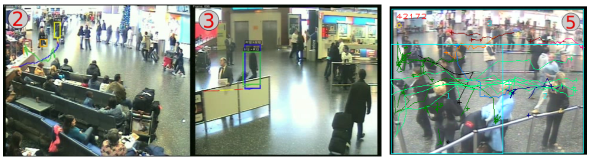
Abstract. *The paper deals with Surveillance Network Augmented by Retrieval (SUNAR) system – an information retrieval based wide area (video) surveillance system being developed as a free software at FIT BUT. It contains both standard and experimental techniques evaluated by NIST at the AVSS 2009 Multi-Camera Tracking Challenge and SUNAR performed comparably well.*
In brief, SUNAR is composed of three basic modules – video processing, retrieval and the monitoring interface. Computer Vision Modules are based on the OpenCV Library for object tracking extended by feature extraction and network communication capability similar to MPEG-7. Information about objects and the area under surveillance is cleaned, integrated, indexed and stored in Video Retrieval Modules. They are based on the PostgreSQL database extended to be capable of similarity and spatio-temporal information retrieval, which is necessary for both non-overlapping surveillance camera system as well as information analysis and mining in a global context.

1 Introduction

Nowadays, there is a lot of data produced by wide area surveillance networks. This data is a potential source of useful information both for on-line monitoring and crime scene investigation. Machine vision techniques have dramatically increased in quantity and quality over the past decade. However, the state of the art still doesn't provide the satisfactory knowledge, except some simple problems such as people counting and left luggage or litter detection.

Justin Davenport in Evening Standard [6] showed statistics of crime-fighting CCTV cameras in Great Britain. The country's more than 4.2 million CCTV cameras caught (in 2007) each British resident as many as 300 times each day. BBC News [1] informed that half a million pounds a year was spent on talking cameras helping to pick up litter. Yet 80% of crime is unsolved. Well, we agree that high quality crime investigation is the best prevention.

The idea was to create an automated system for object visual detection, tracking and indexing that can reduce the burden of continuous concentration on monitoring and increase the effectiveness of information reuse by a security, police, emergency and firemen (or military) and to be useful in the accident investigation. The task is to perform the analysis of the video produced by a camera system with non-overlapping field of views. The analysis, based on cleaned, integrated, indexed and stored metadata, is of two types – on-line used for identity preservation in a wide area; and off-line to query the metadata of the camera records when an accident, crime, a natural or human disaster (war) occurs.



(a) (b)

Fig. 1. An example of a successful camera pair handover.

In 2006, we have started to develop an IR-based multi-camera tracking system to be at the top of the state of the art. We have taken part in several projects (CARETAKER [4]) and evaluations (TRECVid [19]) concerning similar problems. However, the AVSS 2009 Multi-Camera Tracking Challenge [20] was the first evaluation campaign that used the annotated Multiple-camera Tracking (MCT) Dataset from the Imagery Library for Intelligent Detection Systems (i-LIDS) provided by Home Office Scientific Development Branch (HOSDB) of the UK [16]. We have used the MCT video data and annotations to train and evaluate the SUNAR performance and it performed comparably well.

The paper is organized as follows. The introduction presents our motivation and ideas. An overview and design of the SUNAR system is described in the following section. Computer vision methods are described in section 3. Object identification, search and analysis techniques are described in section 4. The NIST performance evaluation of the SUNAR system is in section 5. State of the art is situated at the beginning of each section. The paper is concluded in section 6.

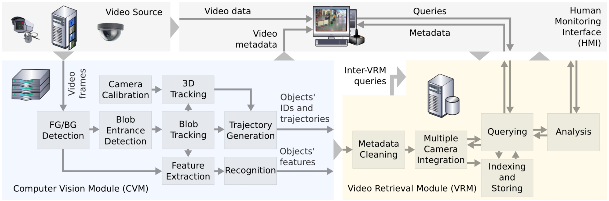


Fig. 2. Illustration of the multiple camera tracking process including the manual annotations

2 System design

Although there are many multi-camera surveillance systems [10, 7, 12, 13], we believe our approach outperforms the others, because those described in literature were not evaluated successfully [10, 12], while those in praxis make many simplifying presumptions (e.g. traffic monitoring). Moreover, in there is no need for a central or primary module [7] or some special hardware such as camera sensors [13]. Moreover, it is able to derive various useful information concerning the entire area under surveillance.

From the schematic perspective, SUNAR consists of the following modules, as illustrated in figure 2:

0. Source of video (any provider)
1. Computer Vision Modules (CVM)
2. Video Retrieval Modules (VRM)
3. Human Monitoring Interfaces (HMI)

The video source might be e.g. a camera or a video server and it is not a generic part of the system. Each module except the Human Monitoring Interface is responsible for capturing, analysis and retrieval in an appropriate part of the wide area under surveillance. Modules communicate basically only with their neighborhoods using the IP protocol. In this way, we can build a considerably large system, because no special central unit is necessary.

The input of the Computer Vision Module (CVM) is a video stream. We use OpenCV [8] for tracking and 3D calibration especially (if feasible). We have extended the OpenCV Blobtrack to be capable of feature extraction, object (and event) recognition and IP based video stream capability. The output of the CVM module is metadata of objects and the environment. It includes local identification of objects, its spatio-temporal location and changes (speed) and a description of objects – dimensions, shape, color, texture or other special features (e.g. license plate and face descriptor) similarly to MPEG-7 [9]. The description is complemented with a recognition of basic object classes (e.g. cars, trolleys, people and groups) and events (e.g. opposing flow and left luggage).

The main contribution of the proposed wide area system is in the Video Retrieval Module (VRM). The input of the module is metadata produced by CVMs. This metadata is cleaned and normalized in time and space (lighting, color bias and 3D parameters) and stored in the PostgreSQL database (www.postgresql.org). The primary function of the VRM is to identify objects – to integrate identifiers (ID) of objects in the wide area, based on the previous occurrence of an object and its appearance. This is accomplished by the use of information retrieval and video search methods based on metadata produced by CVMs as further described in section 4.

The Human Monitoring Interface is then capable not only of a simple monitoring the area, but also querying monitored objects based on their previous occurrences, visual properties and behavior. The behavior is derived from an object's trajectory, its interactions with the environment and mutual interactions based on statistical and data mining methods. This is illustrated in figure 1b.

3 Computer vision techniques

There are two major spheres we would like to evaluate – computer vision and surveillance information retrieval. The computer vision part is further divided in the object tracking, feature extraction and 3D calibration as illustrated in figure 2.

The computer vision is a broad but still underdeveloped area summarized by Sonka, Hlavac and Boyle in [14]. We concern on visual surveillance methods, especially on distributed surveillance systems, reviewed by Valera and Velastin [15] and CARETEKER deliverables [4].

The 3D camera calibration [14] is an optional technique in the IR based approach, when an exact 3D calibration is required, we use CARETAKER's KalibroU a camera calibration program, based on Tsai's method [4]. Thus we concentrate more on tracking, feature extraction and object recognition.

3.1 Object tracking

Object tracking [14] is a complex problem and it is hard to make it working well, in real (crowded) scenes as illustrated in figure 3. Discussed approach is based mainly on proved methods of object tracking implemented in the Open Computer Vision Library [8]. The tracking process is illustrated in figure 2. Background is modeled using Gaussian Mixture Models [8] as an average value of color in each pixel of video and the foreground is a value different to the background. We have been inspired by the approach developed by Carmona et al. [3].

Foreground is derived from background, which is modeled using Gaussian Mixture Models [8] as an average value of color in each pixel of video and the foreground is a value different to the background based on segmentation of the color in RGB color space into background, foreground and noise (reflection, shadow, ghost and fluctuation) using a color difference Angle-Mod cone with vertex located in the beginning of the RGB coordinate system. In this way, the illumination can be separated from the color more easily.

The other two modules blob entrance and tracking are standard OpenCV Blobtrack functions [8]. The blob entrance detection tracks connected components of the foreground mask. The Blob tracking algorithm is based again on connected components tracking and Particle filtering based on Means-shift resolver for collisions. There is also a trajectory refinement using Kalman filter as described in section 4.

The trajectory generation module has been completely rewritten to add the feature extraction and TCP/IP network communication capability. The protocol is based on XML similarly to MPEG-7 [9]. The objects' ID and trajectory is in this way delivered to a defined IP address and service (port 903).

3.2 Feature extraction and object recognition

There are more possibilities how to make a multi-camera surveillance system [7, 12, 13]. Because of our goal – to acquaint as much information about objects as possible, we use visual surveillance information retrieval instead of (multi-)camera homography or handover regions as in [7]. Moreover, the area might be large and objects will occlude in those regions.

Although there are many types of features to be extracted [14], primarily we use descriptors based on the visual part of MPEG-7 [9]. We try to avoid color descriptors only, as in [13], because most of airport passengers (at least on British Isles) wear black coats and there is a lot of dark metallic cars there.

However, we have adopted color layout concept, where each object is resampled into 8x8 pixels in Y'Cb-Cr color model. Then, the descriptor coefficients are extracted zig- zag from its Discrete cosine transform similarly to JPEG. Other (texture) descriptor is based on extraction of energy from (Fourier) frequency domain bands defined by a bank of Gabor filters [9].

For the object classification we use also local features (such as SIFT and SURF) and a simple region (blob) shape descriptor. The shape together with previously described object metadata then acts as an input of a classification algorithm in the recognition procedure of the CVM. The object recognition process is based on 2 popular machine learning methods – AdaBoost and Support vector machines (SVM), the OpenCV [8] implementation. The system has a simple training GUI to mark an object by a simple click while holding a key to associate a blob to its appropriate class or to change the class of a misclassified sample.

To avoid this, CVM may use AdaBoost object detection based on Haar features, similarly to the OpenCV face detection. Unfortunately, there are just a few faces to be detected in the standard TV resolution video and camera setup similar to the MCT dataset. The detector is followed by MPEG-7 Face recognition descriptor [9]. Other face recognition approaches will be compared in the future to allow a more precise and consistent object tracking and recognition in low-resolution images and video. Thus, we concentrate more on retrieval methods at the moment.

4 Surveillance Information Retrieval

Although there were published basics of wide area surveillance systems with non-overlapping fields of view [10], these systems suffer from multiple deficiencies caused by the curse of dimensionality – e.g. they allow only simple handover regions [7] or they are unable to act in a crime investigation process [12, 13], because the real recordings are too massive and of low quality to be analyzed efficiently (as in CSI NY series).

The metadata coming from CVMs local IDs, trajectories and object description must be cleaned, integrated, indexed and stored to be able of querying and analyzing it, as illustrated in figure 2.

4.1 Metadata cleaning

The preprocessed data is supposed to be incomplete or duplicate, biased and noisy. Thus, moving objects are modeled as dynamic systems in which the Kalman filter optimally minimizes the mean of error [5] and it can fill in the missing information (position and velocity) for a few seconds in case the object has been occluded, for instance[1].

At the cleaning step, SUNAR stores metadata representing moving objects and information about the environment under surveillance.

4.2 Indexing and storing

The database model consists of three database schemes in the SUNAR database Process, Training and Evaluation according to their purpose. All schemes contain three main tables that correspond to the fundamental concepts Object, Track and State (as in our former

[1] Available at `www.fit.vutbr.cz/research/view_product.php.en?id=53`

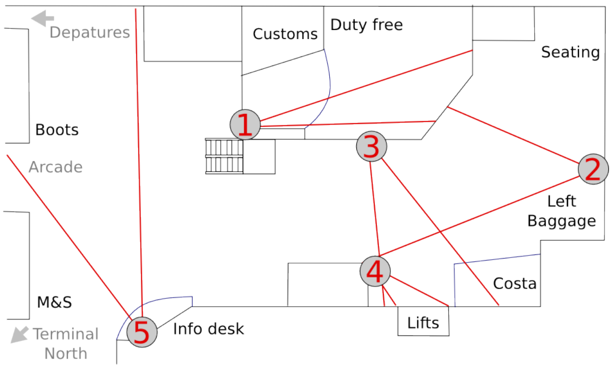


Fig. 3. i_LIDS multiple camera tracking scenario definition provided by HOSDB.

work [5]). Object is an abstract representation of a real object (having a globally unique ID), it is represented by its states. A state consists of two types of features visual properties (as described in section 3) and spatio-temporal features. The latter are represented by location and velocity of an object at a moment. A track is a sequence of such states in a spatio-temporal subspace of the area under surveillance followed by one camera.

The training scheme contains also tables containing statistics and classification models according to the method used. For instance, a simplified Bayesian model table contains columns for source and destination camera IDs, in which objects are passing through. Next columns represent the number of training samples, a prior probability, averages and variances of handover time, trajectory states and visual features. Trajectories are summarized as a weighted average of cleaned states, where the weight is highest at the end of the trajectory. If cameras are overlapping, the handover time may be negative. The average and variance of different feature descriptors acts as the visual bias removal (illlumination, color, viewpoint and blob size calibration) for the integration step.

4.3 Multiple camera integration

The training schema described before is rather simplified. In fact, we use Gaussian Mixture Model and Support Vector Machine [14, 8] models of the (inverted) Kalman filter state as described in our previous work [5]. The inverted state is computed using Kalman filter in the opposite direction the object moved through one camera subspace followed by one camera. The goal of this trick is the classification of the previous subspace (camera) in which it was seen last time most probably.

The object identification then maximizes the (prior) probability of a previous location (camera) multiplied by the normalized similarity (feature distance without bias) to previously identified objects according to average time constraints and visual features in the database [5, 10]. More formally an optimal identifier (k^*) of the object in the wide area is based on its previous occurrence (spatio-temporal, o) and its state (appearance, s):

$$k*(o,s) = argmax_k P(k|o,s) \approx P(o|k)P(s|k) \quad (1)$$

Because of this, we must (approximately) know the camera topology. The figure 3 is suitable enough for the learning step. We have used annotations provided by the HOSDB on i-LIDs MCT dataset. There are 5 cameras and several areas from where a new object can enter.

The object appearance and bias is automatically learned (or summarized) using Gaussian mixture models [8] or optionally SVM. The probability $P(s|k)$ is then determined by a similarity search (the distance is normalized using the sigmoid) with respect to the expected bias, which is simply subtracted.

4.4 Querying

The SUNAR queries are of two types – on-line used for instantaneous condition change and especially for identity preservation as described above; and off-line queries, able to retrieve all the metadata from processed camera records in the wide area after an accident, crime or a disaster happens.

We can distinguish two types of operations: environmental and trajectory operations. Environmental operations are relationships of an objects trajectory and a specified spatial or spatio-temporal environment, such as enter, leave, cross, stay and bypass [2, 5]. Trajectory operations look for relationships of two or more trajectories restricted by given spatio-temporal constraints, such as together, merge, split and visit.

We have also implemented[2] similarity queries based on MPEG-7 features in the PostgreSQL database as a vector (array) distance functions – Eukleidean (Mahalanobis), Chebyshev and Cosine distance.

4.5 Analysis

We perform several types of video analysis, mainly classification and clustering as illustrated in figure 1b. The first type is the modeling based on visual appearance of an object (color layout, blob) using Gaussian Mixture Models (GMM, [8]).

[2] Avaiable at `www.fit.vutbr.cz/research/view_product.php.en?id=73`

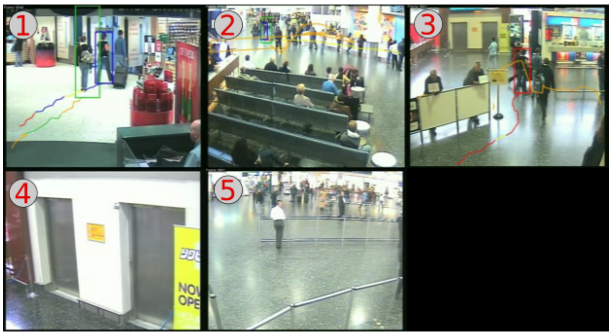
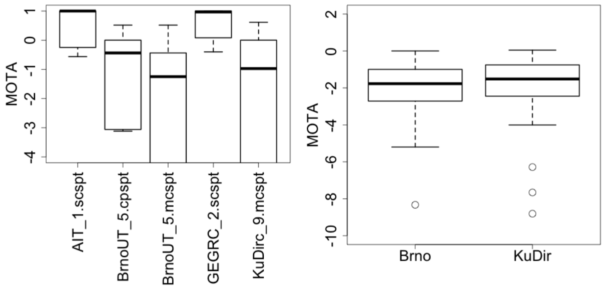


Fig. 4. Illustration of the multiple camera tracking process of the SUNAR system including manual ground truth annotations provided by HOSDB and NIST



(a) (b)

Fig. 5. The NIST's single camera (a) and multiple camera (b) single person tracking MOTA evaluation medians.

Second, we perform trajectory classification based on Gaussian Mixture Models as needed for the multiple camera identification as in section 4 and Hidden Markov Models (HMM). In the article [11] we selected few scenes, where some easily recognizable human behavior occurs. For example, one concept represents if people go through turn pikes or not. The HMM are trained on such classes. The trajectory which doesn't fit any HMM model (with respect to some threshold) is considered to be abnormal. In addition, SUNAR uses velocity and acceleration as training features, which describe and discover some abnormalities better (jump over).

Moreover, using the spatio-temporal queries, we can discover splitting and merging objects, opposing flow (together with GMM and aggregate functions) or an object put (operations enter, split, leave and stay).

5 Evaluation

The previous evaluations such as Performance Evaluation of Tracking and Surveillance (PETS [17]) dealt with other aspects of computer vision than multiple camera surveillance with non-overlapping camera fields of view. They either dealt with classical single camera tracking or they have concerned more on the event detection as Classification of Events, Activities, and Relations. For instance, events so-called left baggage, split, hug, pointing, elevator no entry are detected in the TRECVid Surveillance Event Detection evaluation [19].

The AVSS 2009 Multi-Camera Tracking Challenge [20] was the first evaluation campaign that used the annotated Multiple-camera Tracking (MCT) Dataset from the Imagery Library for Intelligent Detection Systems (i-LIDS) provided by Home Office Scientific Development Branch (HOSDB) in the UK [16]. We have used the MCT video data and annotations to train and evaluate the SUNAR performance. The data set consists of about 44 hours of video recorded by five cameras at the London Gatwick Airport.

The task is defined as: Given 5 in situ video frames with bounding box data specifying a person to be tracked, track the person in 5, 2 or 1 camera views by outputting bounding boxes [20].

We have participated in the compulsory Multi-Camera Single Person Tracking (MCSPT) and Camera Pair Single Person Tracking (CPSPT). The illustration of the data and the area under surveillance is in figures 2, 3 and 1. For more details see [20].

According to Johnatan Fiscus's and Martial Michels's presentation at the 2009 AVSS conference, [20] and received evaluated submissions, they used especially the Multiple Object Tracking Accuracy (MOTA, [18]) metric. The correct detection here is when it states:

$$MOTA = 1 - \frac{\sum_{t=1}^{N_{frames}} (c_m(m^{(t)}) + c_f(fp^{(t)}))}{\sum_{t=1}^{N_{frames}} N_G^{(t)}} \quad (2)$$

The $G_i^{(t)}$ is the ground truth bounding box of an object i at (frame or) time t, the $D_i^{(t)}$ is a (SUNAR) system detection accordingly. Else the detection is false positive $fp^{(t)}$, or missed $m^{(t)}$ if there is no system detection at time t. Then the MOTA is defined as 2. Where c_m and c_f are weights (=1 this time) and N_G is the number of ground-truth objects at time t. The perfect MOTA is 1, but it may go down to $-\infty$ because of false alarms [20]. The (median) MOTA results for single camera and multiple cameras are illustrated in the figure 5. There the camera pair run (BrnoUT_5.cpspt) was better than our multiple camera run (BrnoUT_5.mcspt) because of the state space to be searched. Thus the single camera (scspt) runs are incomparable to multiple camera runs. In table 1, only MCSPT results are depicted.

The table 2 also shows that using standard precision/recall metrics, our results are slightly better than other results [20]. Moreover, using the multiple cam-

MOTA	Brno	KuDir
Test Set Average	-1.183	-1.400
Track Averaged Mean	-2.052	-2.072
Track Averaged Median	-1.770	-1.517

Table 1. Multiple camera tracking results - MOTA.

era (summarized binar) metric the (primary to) Secondary Camera subject Re-Acquisition (SCRA, [20]) shows that SUNAR slightly outperformed the other systems in absolute numbers, which may be seen in table 2. The CPSPT task results were similar to the table above, but we have been the only participants there. The illustration of the task is in figure 1. In both figures 1 and 4 (an illustration of a MCSPT tracking trial), the bounding boxes and trajectories are of five colors. Blue means non- occluding reference (ground truth), yellow an occluding reference. The Green box and trajectory shows a correct detection, red represents a missed detection and the orange color is for false alarms.

		Sec. RA - GT					Sec. RA Brno					Sec. RA KD				
		1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
Primary cam	1	9					1					0				
	2		8					2					0			
	3			7					0					0		
	4		1					0					0			
	5		9					0					0			

Table 2. The primary to Secondary Camera subject Re-Acquisition (SCRA) metric table.

6 Conclusions

This paper presents a state of the art SUNAR surveillance system based on visual information retrieval in praxis (using free software). In contrast to other approaches, we try to collect and index as much information as we can acquaint and manage efficiently to avoid a continuous human CCTV monitoring and analysis of massive and low quality 3 recordings in case of an accident.

The FIT, Brno University of Technology has taken part in many projects and evaluations concerning the public safety and visual surveillance, however the AVSS 2009 Multi-Camera Tracking Challenge [20] was the first public evaluation campaign concerning object tracking in a wide area under surveillance containing both camera setups – overlapping and non-overlapping field of views.

Although we are convinced the system works really good under certain circumstances and it outperformed the others especially in the Secondary Camera subject Re- Acquisition (SCRA) metric at the AVSS conference, there are some issues. Especially those concerning computer vision techniques. It dwells in the object detection, tracking and recognition performance in low quality video and the achievement of the real-time operation.

7 Acknowledgement

The work presented in this paper was funded in part by the FP7 project TA2: Together Anywhere, Together Anytime ICT-2007-214793 and the BUT FIT grant FIT-10-S-2 and the research plan MSM0021630528.

References

1. BBC ’Talking’ CCTV scolds offenders. BBC News, 4 Apr, 2007.
2. S. Brakatsoulas, D. Pfoser, N. Tryfona, Modeling, Storing and Mining Moving Object Databases. IDEAS. 2004.
3. E.J. Carmona, J. Martinez-Cantos and J. Mira.A new video segmentation method of moving objects based on blob-level knowledge. Pattern Recognition Letters, 29, 272-285. 2008.
4. CARETAKER Consortium. Caretaker Puts Knowledge to Good Use. Mobility, The European Public Transport Magazine. 18(13). 2008.
5. P. Chmelar and J. Zendulka. Visual Surveillance Metadata Management. Database and Expert Systems Applications, DEXA ’07. 18th Int. Conf., 79-84. 2007.
6. J. Davenport. Tens of thousands of CCTV cameras, yet 80% of crime unsolved. Evening Standard, 19 Sep, 2007.
7. T. Ellis, J. Black, M. Xu, and D. Makris. A Distributed Multi Camera Surveillance System. Ambient Intelligence. 107-138. 2005.
8. G. R. Bradski. Learning OpenCV. Sebastopol: O’Reilly, 2008. 555 p.
9. ISO/IEC JTC1/SC29/WG11. MPEG-7 Overview. 2004.
10. O. Javed and M. Shah. Automated Visual Surveillance: Theory and Practice. Springer, 110p. 2008.
11. J. Mlich and P. Chmelar. Trajectory classification based on Hidden Markov Models. Proceedings of 18th Int. Conf. on Computer Graphics and Vision, 101-105. 2008.
12. W. Qu, D. Schonfeld and Mohamed M. Distributed Bayesian multiple-target tracking in crowded environments using multiple collaborative cameras. EURASIP J. Appl. Signal Process, (1). 2007.
13. F. Z. Qureshi and D. Terzopoulos. Multi-camera Control through Constraint Satisfaction for Persistent Surveillance. Advanced Video and Signal Based Surveillance, IEEE Conf. on, 211-218. 2008.
14. M. Sonka, V. Hlavac and R. Boyle. Image Processing, Analysis, and Machine Vision, 3rd Edition, Thomson Engineering, Toronto, 800 p. 2007.

15. M. Valera and S.A. Velastin. Intelligent distributed surveillance systems: a review. Vision, Image and Signal Processing, IEE Proceedings, 152(2), 192-204. 2005.
16. HOSDB. Home Office Multiple Camera Tracking Scenario data. scienceandresearch.homeoffice.gov.uk/hosdb/cctv-imaging-technology/video-based-detection-systems/i-lids [cit. 2009-11-17].
17. PETS: Performance Evaluation of Tracking and Surveillance. www.cvg.rdg.ac.uk/PETS2009 [cit. 2009-11].
18. R. Kasturi et. al. Framework for Performance Evaluation of Face, Text, and Vehicle Detection and Tracking in Video: Data, Metrics, and Protocol, IEEE Trans. on Pattern Analysis and Machine Intelligence. vol. 31, no. 2, pp. 319-336. February 2009.
19. TRECVid Event Detection. www-nlpir.nist.gov/projects/tv2009/tv2009.html#4.1 [cit. 2009-11-17].
20. J. Fiscus and M. Michel. AVSS 2009 Multi-Camera Tracking Challenge www.itl.nist.gov/iad/mig/tests/avss/2009/index.html [cit. 2009-11-17].