Computer Vision and Image Understanding 116 (2012) 320–329

Contents lists available at SciVerse ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Multi-scale and real-time non-parametric approach for anomaly detection and localization ☆

Marco Bertini *, Alberto Del Bimbo, Lorenzo Seidenari

Università degli Studi di Firenze – MICC, Firenze, Italy

ARTICLE INFO

Article history:
Received 14 March 2011
Accepted 1 September 2011
Available online 25 October 2011

Keywords:
Video surveillance
Anomaly detection
Space–time features

ABSTRACT

In this paper we propose an approach for anomaly detection and localization, in video surveillance applications, based on spatio-temporal features that capture scene dynamic statistics together with appearance. Real-time anomaly detection is performed with an unsupervised approach using a non-parametric modeling, evaluating directly multi-scale local descriptor statistics. A method to update scene statistics is also proposed, to deal with the scene changes that typically occur in a real-world setting. The proposed approach has been tested on publicly available datasets, to evaluate anomaly detection and localization, and outperforms other state-of-the-art real-time approaches.

1. Introduction and previous work

The real-world surveillance systems currently deployed are primarily based on the performance of human operators that are expected to watch, often simultaneously, a large number of screens (up to 50 [2]) that show streams captured by different cameras. One of the main tasks of security personnel is to perform proactive surveillance to detect suspicious or unusual behavior and individuals [3] and to react appropriately. As the number of CCTV streams increases, the task of the operator becomes more and more difficult and tiring: after 20 min of work the attention of an operator degrades [4]. Operators usually take into account specific aspects of activity and human behavior in order to predict possible perilous events [2], although often they can not explain their own criteria used to detect an unusual situation [3], or do not recognize unusual behaviors because they have not gathered enough knowledge of the environment and of the common behaviors they have to watch [5].

Video analytics techniques that automatically analyze video streams to warn, possibly in real-time, the operators that unusual activity is taking place, are receiving much attention from the scientific community in recent years. The detection of unusual events can be used also to guide other surveillance tasks such as human behavior and action recognition, target tracking, and person and car identification; in this latter case it is possible to use pan-tilt-zoom cameras to capture high resolution images of the subjects that caused the anomalous events.

Anomaly detection is the detection of patterns that are unusual with respect to an established normal behavior in a given dataset, and is an important problem studied in several diverse fields [6]. Approaches to anomaly detection require the creation of a model of normal data, so to detect deviations from the model in the observed data. Three broad categories of anomaly detection techniques can be considered, depending on the approach used to learn the model: supervised [7–14], semi-supervised [15,16] or unsupervised [17–28]. In this work we follow an unsupervised approach, based on the consideration that anomalies are rare and differ amongst each other with unpredictable variations.

The model can be learned off-line as in [7,8,10,29] or can be incrementally updated (as in [19,20,22,26]) to adapt itself to the changes that may occur over time in the context and appearance of a setting. Our approach continuously updates the model, to gather knowledge of common events and to deal with changes in "normal" behavior, e.g. due to variations in lighting and scene setting.

Most of the methods for identifying unusual events in video sequences use trajectories [8–10,13,15–17,23,28–30] to represent the activities shown in a video. In these approaches objects and persons are tracked and their motion is described by their spatial location. Blob features have been used in [20,27,31], without tracking the blobs. The main drawback of tracking-based approaches is the fact that only spatial deviations are considered anomalies, thus abnormal appearance or motion of a target that follows a "normal" track is not detected.

Optical flow has been used to model typical motion patterns in [11,19,21,22,31], but, as noted in [29], this measure also may become unreliable in presence of extremely crowded scenes; to solve this issue a dense local sampling of optical flow has been adopted

☆ An earlier version of this paper appeared in a conference proceeding [1].

* Corresponding author.

E-mail address: bertini@dsi.unifi.it (M. Bertini).

1077-3142/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2011.09.009

in [12,19]. Local spatio-temporal descriptors have been successfully proposed in [32,33] to recognize human actions, while more simple descriptors based on spatio-temporal gradients have been used to model motion in [18,29] for anomaly detection. Dynamic textures have been used to model multiple components of different appearance and dynamics in [25,34].

Another issue that is common to both tracking and blob-based approaches is the fact that it is very difficult to cope with crowded scenes, where precise segmentation of a target is impossible. It is also important to consider that trajectory based methods rely on a long chain of algorithms (blob detection, data association, tracking, ground plane trajectory extraction) each of which may fail, leading to the failure of the whole anomaly detection system. Instead, approaches that are purely pixel-based, learning a scene representation independently of the explicit modeling of object motion, allow to skip the chain of intermediate decisions required by the chain of algorithms, and detect an event directly from the representation of frames.

Some recent works consider the fact that, in some cases, an event can be regarded as anomalous if it happens in a specific context; for example the interaction of multiple objects may be an anomaly even if their individual behavior, if considered separately, is normal. These works consider the scene [27,22,29], typically modeled with a grid of interest points, or the co-occurrence of behaviors and objects [14,21,25,28] like persons and vehicles.

In this work we propose a multi-scale non-parametric approach that detects and localize anomalies, using dense local spatio-temporal features that model both appearance and motion of persons and objects. Real-time performance is achieved using a careful modeling of dense sampling of overlapping features. Using these features it is possible to cope with different types of anomalies and crowded scenes. The proposed approach addresses the problem of high variability in unusual events and, using a model updating procedure, deals with scene changes that happen in real world settings. The spatial context of the spatio-temporal features is used to recognize contextual anomalies.

The rest of this paper is structured as follows: scene representation, spatio-temporal descriptor and feature sampling are described in Section 2; in Section 3 is presented the real-time anomaly detection method, with multi-scale integration, context modeling and model updating procedure; finally experimental results, obtained using standard datasets are discussed in Section 4. Conclusions are drawn in Section 5

2. Scene representation

Modeling crowd patterns is one of the most complex contexts for detection of anomalies in video surveillance scenarios. Describing such statistics is extremely complex since, as stated in Section 1, the use of trajectories does not allow to capture all the possible anomalies that may occur, e.g. due to variations of scene appearance and the presence of unknown objects moving in the scene; this is due to the fact that object detection and tracking are often unfeasible both for computational issues and for occlusions. On the other hand, global crowd descriptors are not able to describe anomalous patterns which often occur locally (e.g. a cyclist or a person moving in an unusual direction among a crowd). The most suitable choice in this context is to observe and collect local space–time descriptors.

2.1. Feature sampling

Surveillance scenes are typically captured using low frame rate cameras or at a distance, leading to a short temporal extent of actions and movements (often just 5–10 frames). Therefore, it is necessary to sample these features densely in order to obtain as complete as possible coverage of the scene statistics. This approach is also motivated by the good performance obtained using dense sampling in object recognition [35] and human action recognition [36].

The solution adopted in this work is to use spatio-temporal features that are densely sampled on a grid of cuboids that overlap in space and time. Fig. 1 shows an example of spatial, temporal and spatio-temporal overlaps of cuboids, and an example of application of overlapping spatio-temporal cuboids to a video. This approach permits localization of an anomaly both in terms of position on the frame and in time, with a precision that depends on the size and overlap of cuboids; it also models the fact that certain parts of the scene are subject to different anomalies, illumination conditions, etc., and is well suited for the typical surveillance setup where a fixed camera is observing a scene over time. Considering the position of the cuboids on the grid it is also possible to evaluate the context of an anomaly, inspecting the nearby cuboids. Moreover, it makes it possible to reach real-time processing speed, since it does not require spatio-temporal interest point localization. In our previous work [1] we have investigated how the overlap affects the performance of the system, and determined that a 50% spatial overlap provides the best performance, detecting more abnormal patterns without raising false positives, because spatial localization of the anomaly is improved. On the other hand temporal overlap does not provide an improvement and, instead, may increase false detections.

2.2. Spatio-temporal descriptors

To compute the representation of each spatio-temporal volume extracted on the overlapping regular grid, we define a descriptor based on three-dimensional gradients computed using the luminance values of the pixels (Fig. 1). Each cuboid is divided in subregions. Each subregion is described by spatio-temporal image gradient represented in polar coordinates as follows:

$$M_{3D} = \sqrt{G_x^2 + G_y^2 + G_t^2}, \tag{1}$$

$$\phi = \tan^{-1}\left(G_t / \sqrt{G_x^2 + G_y^2}\right), \tag{2}$$

$$\theta = \tan^{-1}(G_y/G_x), \tag{3}$$

where G_x, G_y and G_t are computed using finite difference approximations:

$$G_x = L_{\sigma_d}(x+1,y,t) - L_{\sigma_d}(x-1,y,t), \tag{4}$$

$$G_y = L_{\sigma_d}(x,y+1,t) - L_{\sigma_d}(x,y-1,t), \tag{5}$$

$$G_t = L_{\sigma_d}(x,y,t+1) - L_{\sigma_d}(x,y,t-1). \tag{6}$$

L_{σ_d} is obtained by filtering the signal I with a Gaussian kernel of bandwidth σ_d to suppress noise; in all the experiments we have used $\sigma_d = 1.1$, a value which proved to be effective in representing space–time patches in our previous work in human action recognition [37]. We compute two separate orientation histograms by quantizing ϕ and θ and weighting them by the magnitude M_{3D}.

It can be observed that if the overlap of cuboids precisely matches the subregions of nearby cuboids we can reuse the computations of these subregions for different cuboids' descriptors (Fig. 2). Using a number of spatial subregions that is a multiple of the overlap reduces the computational cost of the descriptors [38]: considering that a 50% overlap of cuboids is optimal then it is convenient to use an even number of spatial regions, since it is possible to reuse 50% or, depending on the position of the cuboid, 75% of the descriptors of nearby cuboids.

Therefore, we have divided the cuboid in 8 subregions, two along each spatial direction and two along the temporal direction.

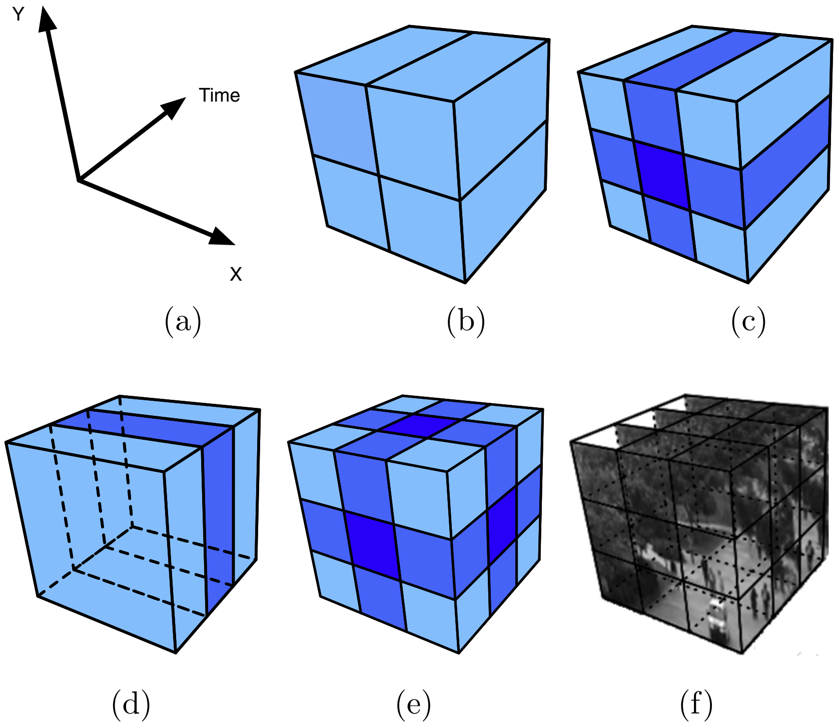

Fig. 1. Examples of cuboids for spatio-temporal descriptors extraction, darker areas show the common parts due to the overlap of cuboids, if any. (a) spatial dimensions (*X* and *Y*) and temporal dimension (Time); (b) 2 × 2 × 1 cuboids with no overlap; (c) 2 × 2 × 1 cuboids with spatial overlap and no temporal overlap; (d) 1 × 1 × 2 cuboid with temporal overlap and no spatial overlap; (e) 2 × 2 × 2 cuboids with spatio-temporal overlap; (f) 2 × 2 × 2 cuboids with spatio-temporal overlap, applied to a part of a frame of a surveillance video, to compute the spatio-temporal descriptors.

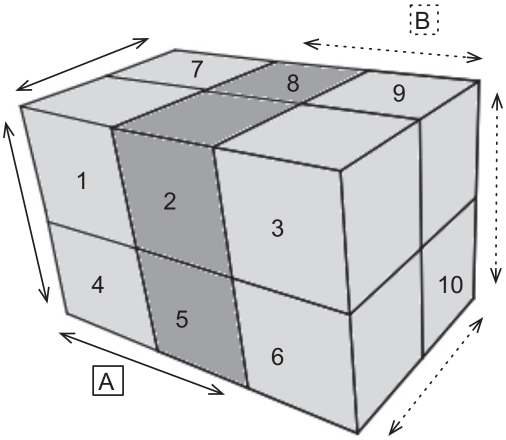

Fig. 2. Example of two overlapping cuboids: (A and B) The subregions 2, 5, 8 (and another one below 8) are common to both cuboids, and their computation for cuboid B can skipped, once they have been computed for cuboid A.

This choice increases the speed of the system of about 50%, with respect to a division of cuboids in 3 × 3 × 2 regions [1].

This descriptor jointly represents motion and appearance, and it is robust to illumination and lighting changes, as required in a surveillance context in which a video might be recorded over a large extent of time. We do not apply a re-orientation of the 3D neighborhood, since rotational invariance, otherwise useful in object detection and recognition tasks, is not desirable in a surveillance setting. The ϕ (with range $-\frac{\pi}{2},\frac{\pi}{2}$) and θ $(-\pi, \pi)$ are quantized in four and eight bins, respectively. The overall dimension of the descriptor is thus $2 \times 2 \times 2 \times (8+4) = 96$. Fig. 3 shows three descriptors of cuboids containing a walking person, a cyclist and a moving cart.This construction of the three-dimensional histogram is inspired, in principle, by the approach proposed by Scovanner et al. [39], where they construct a weighted three-dimensional histogram normalized by the solid angle value (instead of separately quantizing the two orientations) to avoid distortions due to the polar coordinate representation. However, we have found that our method is computationally less expensive, equally effective in describing motion information given by appearance variation, and shows an accuracy of human action recognition that is above or in line with other state-of-the-art descriptors [40], but without requiring tuning of descriptor parameters. In fact, we cannot afford any descriptor parameter learning since our setting is completely unsupervised.

3. Real-time anomaly detection

Our system is able to learn from a normal data distribution fed as a training set but can also start without any knowledge of the scene, learning and updating the “normal behavior” profile dynamically, almost without any human intervention. The model can always be updated with a very simple procedure. Despite the simple formulation of this approach our system is able to model complex and crowded scenes, including both dynamic and static patterns.

Our technique is inspired by the one proposed in [24], where the proposed scene representation is global and static, based on global histograms of oriented gradients of single frames. Instead, in our approach, we use local spatio-temporal features as a scene representation and we exploit the idea of the adaptive threshold in order to learn, over time, local models for different portions of the scene. Another significant difference with respect to [24] is the use of pure data instead of clusters. We do not perform clustering on data since we prefer not to corrupt data distribution in order to produce a more accurate estimation of the distance threshold used to detect anomalies. Also the model update procedure is different: since we are not applying any clustering procedure to

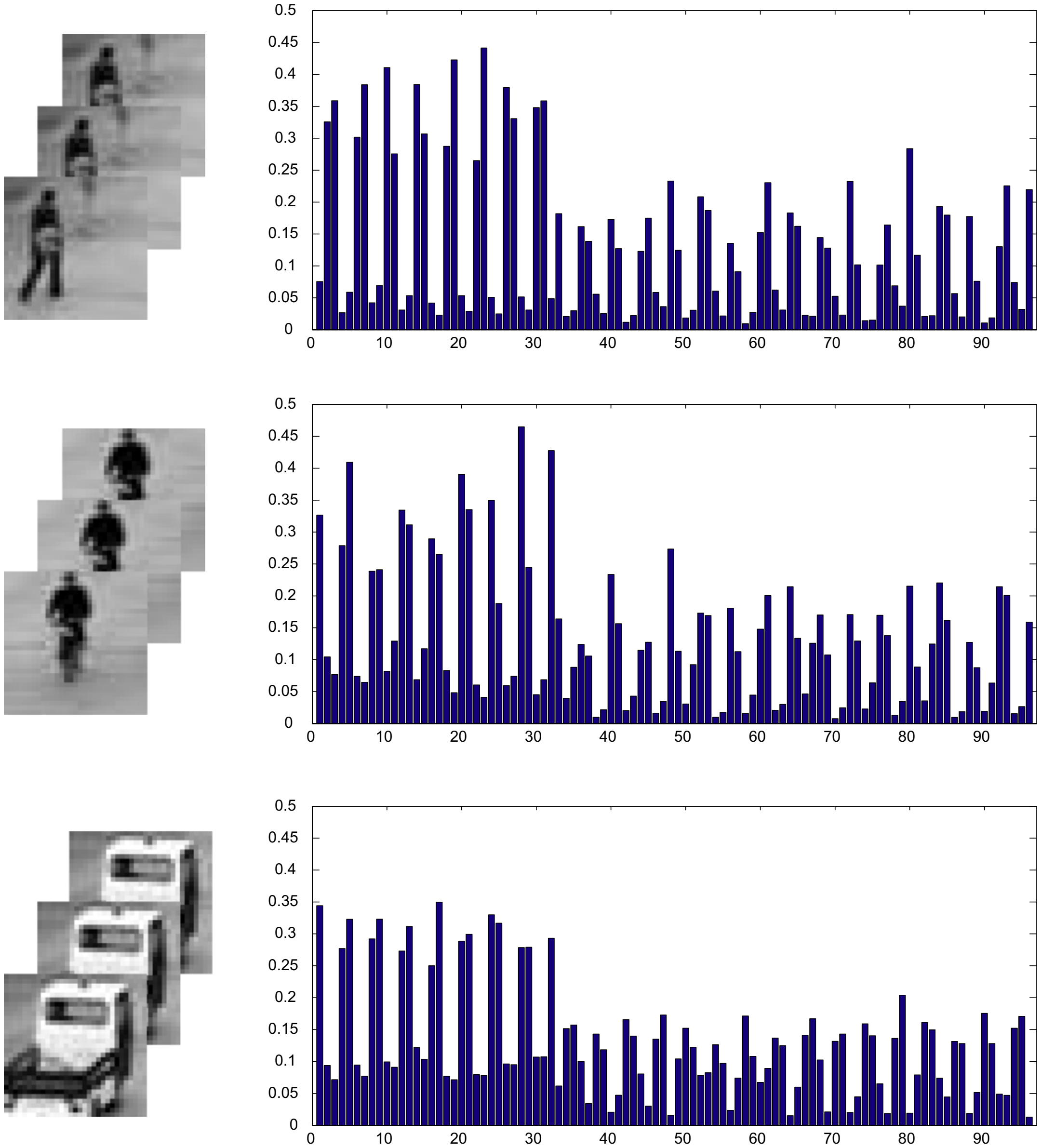

Fig. 3. Example of three descriptors computed on cuboids containing a moving person, a cyclist and a moving cart.

data, our model update can be performed just by analyzing the detected anomalies stored over time; therefore it can be performed more frequently, without the need to operate either in detection mode or in maintenance mode.

As specified in Section 2 the use of local space–time gradients allows us to detect a wider range of anomalies while an appearance based method restricts the anomalies that can be detected only to significant changes in a scene, e.g. a car parked in a wrong place, the presence of a fire truck or an unseen weather condition (rain, snow or fog).

3.1. Non-parametric model

In anomaly detection tasks a certain amount of normal data is usually available; our system can exploit this data as a training set to bootstrap itself and run in a semi-supervised fashion. Our system can also be run on-line with no previous knowledge of the scene, since a model update procedure is used. To jointly capture scene motion and appearance statistics we use the robust space–time descriptor, with dense sampling, described in Section 2. In order to decide if an event is anomalous we need a method to estimate normal descriptor statistics. Moreover, since no assumptions are made on the scene geometry or topology, it is important to define this normal descriptor distribution locally with respect to the frame.

Given a set of triples composed of descriptors d_q, their locations l_q and their scales s_q extracted from the past T frames, we would like to evaluate the likelihood of this data given the previously observed triples $\langle d,l,s \rangle$, i.e. $p(d_q,l_q,s_q|\mathbf{d},\mathbf{l},\mathbf{s})$. The following assumptions are made: descriptors computed from neighboring cells and from cells extracted at different scales are considered independent: this is a common Markovian assumption in low-level vision [41] that, even if may not hold for overlapping cells, allows to simplify the model and indeed proved to be effective, as reported in the experiments. We do not pose any prior on the locations, i.e. we do not consider any region of the frame more likely to generate anomalous descriptors. Since we consider a sequence of frames anomalous if at least a cell of the frame is considered as such, then the whole frame probability is obtained by marginalizing out the cell locations i. In the case of a single scale model we have:

$$p(d_q, l_q, s_q|\mathbf{d}, \mathbf{l}, \mathbf{s}) \propto \sum_i p\left(d_q^i, l_q^i|\mathbf{d}^i, \mathbf{l}^\mathbf{i}\right). \tag{7}$$

For multi-scale models, we assume descriptors computed at different scales independent (even if overlapped), therefore we obtain:

$$p(d_q, l_q, s_q|\mathbf{d}, \mathbf{l}, \mathbf{s}) \propto \sum_i \prod_{j\in O^i} p\left(d_q^i, l_q^i, s_q^i|\mathbf{d}^\mathbf{i}, \mathbf{l}^\mathbf{i}, \mathbf{s}^\mathbf{j}\right), \tag{8}$$

where O^i represent the set of patches overlapping region i.

To model the contextual anomalies, we need to compute the likelihood of a given descriptor with respect to its neighboring observed cells; since we consider neighboring models independent we obtain the following likelihood:

$$p(d_q, l_q, s_q|\mathbf{d}, \mathbf{l}, \mathbf{s}) \propto \sum_i \prod_{j\in O^i} \prod_{k\in N^{ij}} p\left(d_q^i, l_q^i, s_q^i|\mathbf{d}^\mathbf{k}, \mathbf{l}^\mathbf{k}, \mathbf{s}^\mathbf{j}\right), \tag{9}$$

where N^{ij} represents the set of neighboring locations at the same scale. The evaluation of probabilities in Eqs. (7)–(9) are performed through non-parametric tests, as described in the following.

3.2. Implementation

Given a certain amount of training frames for each cell in our grid, space–time descriptors are collected and stored using a structure for fast nearest-neighbor search, providing local estimates of anomalies; an overview of this schema is shown in Fig. 4. The training stage is very straightforward, since we do not use any parametric model to learn the local motion and appearance; instead we represent scene normality directly with descriptor instances.

A simple way to decide if an event happening at a certain time and location of the video stream should be considered anomalous, is to perform a range query on the training set data structure to look for neighbors. In this work we have used a fast approximate nearest-neighbor search over k-means trees, provided by the FLANN library [42]. A k-means tree is a hierarchical indexing data structure obtained by recursively splitting data. Once an optimal radius for each image location is learned, all patterns for which the range query does not return any neighbor are considered anomalies. The problem with this technique is the intrinsic impossibility of selecting *a priori* a correct value for the radius. This happens for two reasons: firstly, each scene location undergoes different dynamics, for example a street will mostly contains fast unidirectional motion generated by cars and other vehicles, while a walkway will have less intense motion and more variations of the direction; moreover a static part of the scene, like the side of a parking lot, will mostly contain static information. Secondly, we want to be able to update our model dynamically by adding data which should be considered normal given the fact that we observed that kind of pattern for a sufficient amount of time; therefore, since scene statistics must evolve over time, the optimal radius will evolve too. Finally, we also would like to select a value that encodes the system sensitivity, i.e. the probability that the observed pattern is not generated from the underlying scene descriptors distribution.

To estimate the optimal radius for each data structure we exploit CDF_i^{-1}, the inverse of the empirical cumulative distribution of nearest-neighbor distances of all features in the structure of the cell i of the overlapping grid (Fig. 5 shows an example for two grid cells). The estimate of the CDF of a random variable d for a value t is:

$$CDF(t) = \sum_{i=1}^{n} \mathbf{1}\{d_i \leqslant t\}, \tag{10}$$

where $\mathbf{1}\{E\}$ is the indicator of event E and d_i are realizations of d. A practical and efficient procedure to directly estimate the inverse empirical cumulative distribution CDF^{-1} of a set D of realizations of univariate random variables d_i, which share their density function, is the following: (1) sort d_i in ascending order, (2) remove duplicate values from the sorted list (usually needed for discrete variables) and store the sorted unique values in a vector D^{su}, (3) obtain $CDF^{-1}(p) = D_k^{su}$, where $k = \lfloor p \cdot |D^{su}| \rfloor$, p is a probability, $|D^{su}|$ is the cardinality of the set D^{su} and v_i denote the ith element of vector v.

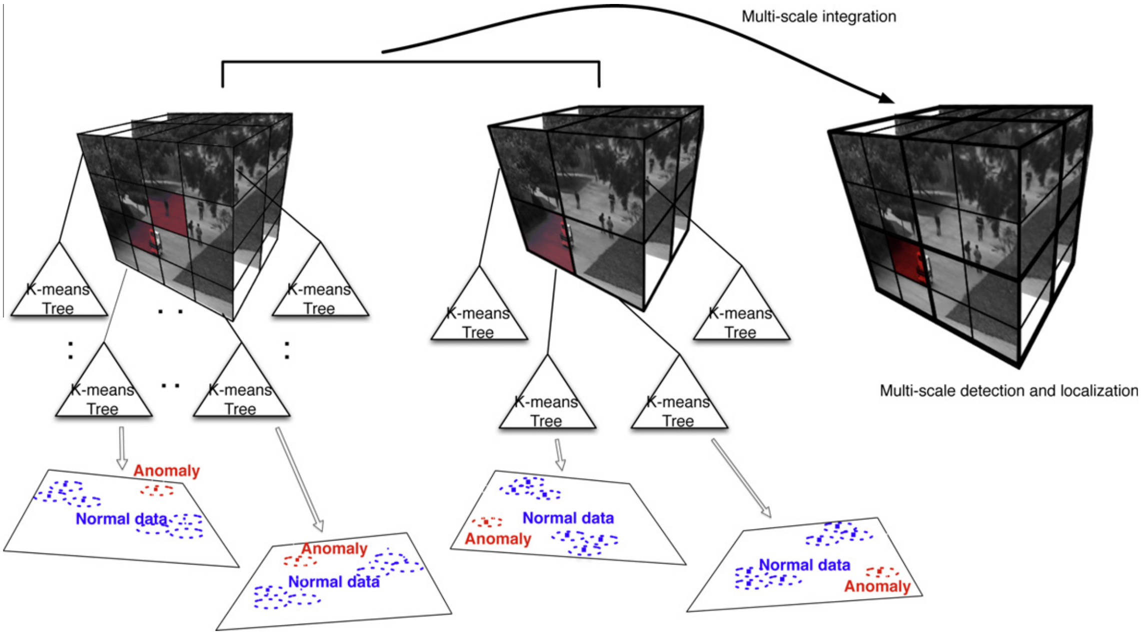

Fig. 4. System overview. For each cell at each scale cuboids features are stored in efficient indexes based on k-means tree (fine on the left, coarse on the right). The planes underneath represent in a simplified view the high dimensional feature space. Anomaly detection may occur in different cells depending on the scale; the multi-scale integration mechanism reduces false alarms and provides a refined localization of the anomaly (e.g. the cart on the walkway, see Fig. 6).

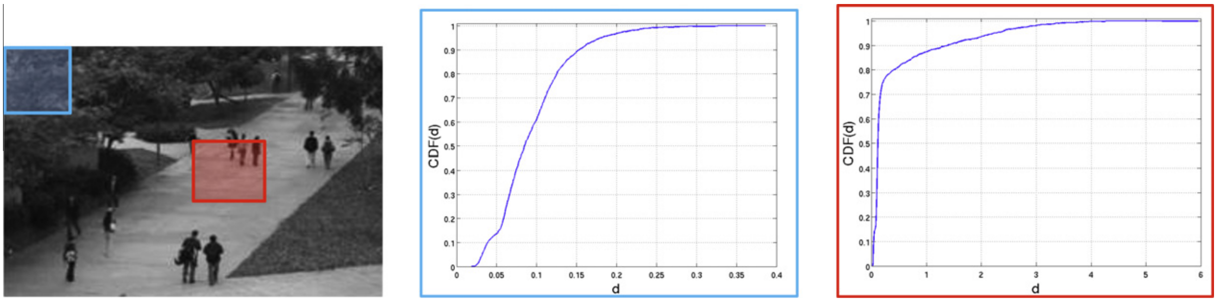

Fig. 5. CDFs of different spatio-temporal cells: *left)* frame with highlighted positions of two cells, *center)* CDF of the blue (upper left) cell, *right)* CDF of the red (centered) cell.

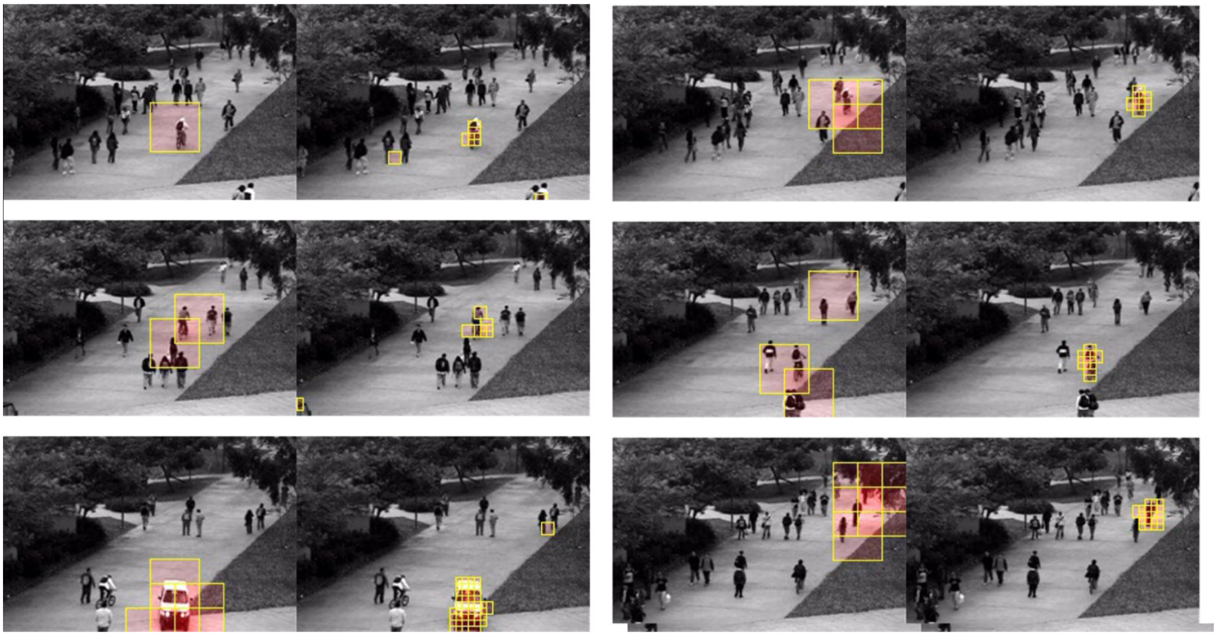

Fig. 6. Single scale anomalies detections and localizations, before integration.

Given a probability p_a below which we consider an event anomalous, we choose the radius $\hat{r}_i$ for cell i as:

$$\hat{r}_i = CDF_i^{-1}(1 - p_a). \tag{11}$$

The anomaly probability p_a can be set to $10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, \ldots$ depending on the user's need to obtain a more or less sensitive system. After setting such value p_a, optimal radii are estimated for each cell with likely different values. This optimal radius formulation allows easy data-driven parameter selection and model update.

3.3. Multi-scale integration

Anomalous events are generated by objects moving in different parts of the scene, therefore their scale can be subject to high variations due to the distance from the camera; moreover, we do not know the kind and the size of the objects that will generate anomalies. It is thus necessary to analyze video at multiple scales. In some initial experiments we observed that models based on smaller patches have a higher segmentation accuracy but suffer from false positives, while for models with bigger patches we observe the opposite behavior. We propose to improve our previous work [1] by exploiting a late fusion of the detection results of multiple models. This captures the abnormal patterns at different resolution. Since we aim at real-time performance, a dense patch sampling in scale is not computationally feasible; therefore, we limit the use of scales to two levels. Models are trained with patches of different size, with a factor of 4× difference. Anomaly detection is performed using the radius search with the optimal learned distance and the final detection result is obtained from the intersection of all the detections. This allows the system to filter spurious small false positives and increases the capability of the system to accurately localize even smaller objects (i.e. pedestrians, cyclists). Moreover, space–time patterns that span more than one overlapping cell will be more likely considered anomalous, while a single isolated patch will be suppressed by the integration procedure. From a probabilistic point of view two likelihood maps are generated non-parametrically. These maps represent how likely it is that a given space–time pattern it is an outlier for the observed statistic; the final likelihood map is generated via a product rule, resulting in the spatial intersection of the two detected areas. In our implementation, in order to keep the system executing in real-time, we used the following scales: 40×40 and 10×10, with a 50% overlap of 20 pixels and 5 pixels, respectively. Fig. 6 shows different anomaly localizations at these scales.

3.4. Context modeling

A purely data-driven method, as the approach proposed in this work, can suffer from the lack of data in the case that statistics of patches from a region is too complex. This is a well known problem in all instance based methods like k-NN. To moderate this effect, we extend our model by considering the anomaly likelihood of a patch with respect to the observations of the nearby patches. Therefore we test the patch descriptor also against the models of the eight neighboring cells. With this technique we increase the amount of data available for learning the local model of a part of the scene in a sensible way; in fact a patch that it is anomalous for a region but not for the neighboring ones would not be considered as such, while patches that are outliers for all the neighboring regions will be considered anomalies. The result of the detection is again obtained by product rule, therefore a patch is anomalous if and only if it is evaluated as such by all the models in its neighborhood.

3.5. Model update

Since applications for anomaly detection in video surveillance are designed to be executed for a long time, it is very likely that a scene will change its appearance over time; very simple examples are the event of a snowstorm, the cars that enter and exit a parking lot or the placement of temporary structures in a setting. It is therefore very important to provide a way to update our model. Again, we propose a very straightforward data-driven technique.

Together with the data-structure for each overlapping grid cell, we keep a list of anomalous patterns. We exploit the same range query approach presented in the previous subsection to look for normality in the abnormality list. This list is inspected on a regular basis, and new data is incorporated by applying the following procedure. If an event happens very frequently it is likely that it will a have certain amount of neighbors in feature space, while truly anomalous event will still be outliers. After the estimation of an optimal radius for the anomalous pattern list, we discard all outliers in this list and incorporate all other data in the cell i training set. The optimal radius $\hat{r}_i$ for the updated cell is then recomputed.

Even if it is not required, since they can be used with default values, two parameters of the system can be tuned to adapt them to a particular scenario: grid density and overlap of cuboids. Reducing cuboid overlap can increase the detection performance, while using a more or less dense spatio-temporal grid can serve also as a system adaptation for a specific camera resolution or frame rate. These two parameters are directly bound to physical and technical system properties (e.g. camera resolution and computer processing speed) that the user can easily adjust to figure out a proper configuration. Instead, the system automatically computes the optimal radius parameter, that is a quantity that is extremely task, scene and time dependent.

4. Experimental results

We tested our approach on the UCSD[1] anomaly dataset presented in [34], which provides frame-by-frame local anomaly annotation. The dataset consists of two subsets, corresponding to different scenes using fixed cameras that overlook pedestrian walkways: one (called Peds1) contains videos of people moving towards and away from the camera, with some perspective distortion; the other (called Peds2) shows pedestrian movement parallel to the camera. Videos are recorded at 10 FPS with a resolution of 238 × 158 and 360 × 240, respectively. This dataset mostly contains sequences of pedestrians in walkways; annotated anomalies, that are not staged, are related to appearance and behavior. In particular, they are non-pedestrian entities (cyclists, skaters, small carts) accessing the walkway and pedestrians moving in anomalous motion patterns or in non-walkway regions. The first subset contains 34 training video samples and 36 testing video samples, while the latter contains 16 training video samples and 12 testing video samples. Each sequence lasts around 200 frames, for a total dataset duration of ~33 min. 10 videos of the Peds1 subset have manually generated pixel-level binary masks, which identify the regions containing anomalies. We tested our approach on the whole UCSD dataset. Each anomalous frame in the testing set is annotated; for each

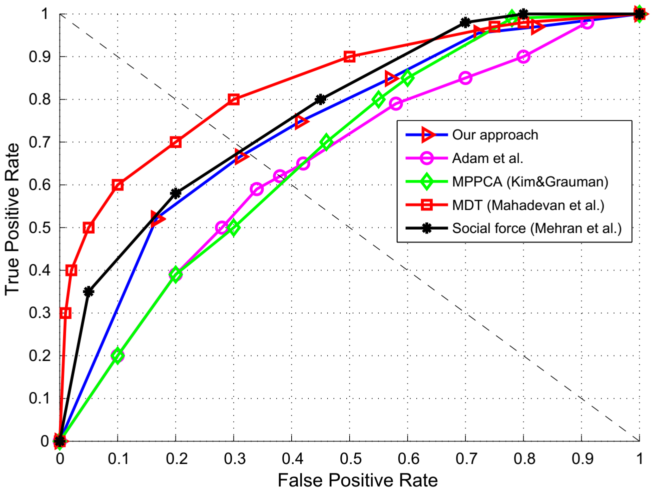

Fig. 7. ROC curve to compare our method with state-of-the-art approaches on the Peds1 dataset. The dashed diagonal is the EER line.

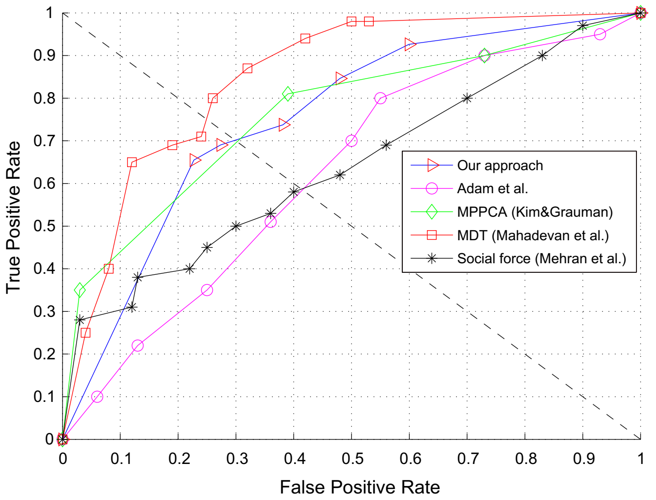

Fig. 8. ROC curve to compare our method with state-of-the-art approaches on the Peds2 dataset. The dashed diagonal is the EER line.

Table 1
Summary of quantitative system performance and comparison with state-of-the-art (lower values are better). EER is reported for frame level anomaly detection on Peds1 and Peds2 datasets together with the average over the two datasets.

	UCSPed1 (%)	UCSPed2 (%)	Average (%)
Single scale	34	32	33
Multi-scale	32	31	32
Context + multi-scale	31	30	30
MDT [34]	25	25	25
MPPCA [22]	40	30	35
Social Force [21]	31	42	37
Adam et al. [19]	38	42	40

[1] http://www.svcl.ucsd.edu/projects/anomaly/dataset.htm.

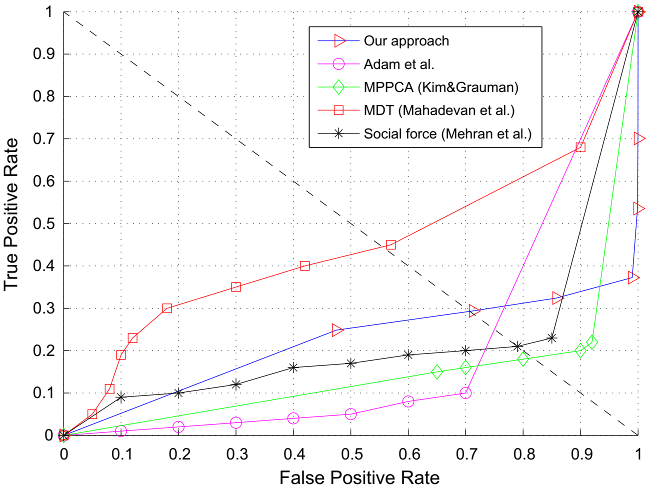

Fig. 9. ROC curve to compare the localization accuracy of our method with state-of-the-art approaches using Peds1 dataset. The dashed diagonal is the EER line (note that the plot of a random classifier is not diagonal in this case, but close to zero).

cuboid classified as anomalous, we flag as anomalous each region of the frames from which it was created; frames that contain at least one anomalous region are considered anomalous. We follow the evaluation procedure of [34]: in the frame level evaluation an abnormal frame is considered correctly detected if at least one pixel of the frame is detected as anomalous; in the pixel level evaluation an abnormal frame is considered correctly detected only if at least the 40% of the anomalous pixels are detected correctly and considered a false positive otherwise. A "lucky guess" happens when a region different from the one that generated the anomaly is detected as anomalous in the same frame. The frame level detection evaluation does not takes into account this phenomenon. In our previous work [1] we evaluated the best parameters for dense sampling and overlapping of the spatio-temporal descriptors; the best results were obtained for cuboids of 40 × 40 pixels, with 8 frames of depth, a spatial overlap of 50% and no temporal overlap. In these experiments we used the same parameters.

We compare our system with results of other state-of-the-art approaches, as reported in [34]: MPCCA [22], Adam et al. [19], Mehran et al. [21] and Mahadevan et al. [34]. Results are reported using the ROC curve and the Equal Error Rate (EER) – that is the rate at which both false positives and misses are equal. Both multi-scale integration and contextual modeling help in lowering the EER, with respect to our previous work [1]. Our approach achieves a similar performance on both Peds1 and Peds2 datasets, showing the flexibility of the representation that is able to cope with diverse settings and types of anomalies. The other competing real-time approaches have a EER performance in the two datasets that varies between 6% and 11%; it can also be noted that these performance variations of the other systems are not uniform, thus there is no hint that a dataset is "more difficult" than the other. Figs. 7 and 8 and Table 1 report the results for anomaly detection in Peds1 and Peds2. Fig. 9 and Table 2 report results for anomaly localization on Peds1.

Our approach, with the use of multiple scales and contextual queries, obtains the second best result both in temporal and spatial anomaly detection after the method proposed in [34], and is far superior to all the others in terms of spatial localization and frame level localization (except the close result of Social Force for Peds1). However, it has to be noted that the approach of [34] is not suitable for real-time processing since it takes 25 s to process a single frame on a computer with a computational power (3 GHz CPU with 2 GB of RAM) comparable to the one used in our experiments (2.6 GHz CPU with 3 GB of RAM). The good results in spatial anomaly localization imply that we are not taking advantage of lucky guesses, but that we accurately localize the abnormal behaviors in space and time. Fig. 10 shows a qualitative comparison of anomaly localization of our approach with state-of-the-art off-line approach [34].

Since our approach aims at real-time processing, we have evaluated the impact of the dense sampling of cuboids, computing the average number of processed frames per second while varying the spatial overlap of cuboids. The plot in Fig. 11 shows how the steps of our method affect the performance. The use of multiple scales degrades the performance the most, almost halving the frame rate. The overhead of context modeling depends on the amount of features extracted, in particular it has little influence for the single scale algorithm but it strongly affects the multi-scale one since the complexity of that step depends linearly on the amount of feature extracted. We also measured the anomaly detection overhead by computing the different frame rate in training (i.e. feature extraction and computation only) and testing and we found that for the single scale approach, without exploiting the context, it is

Table 2
Detection rate on the anomaly localization task (higher values are better).

Single scale	Multi-scale	Context + multi-scale	MDT	MPPCA	SF	Adam
27%	28%	29%	45%	18%	21%	24%

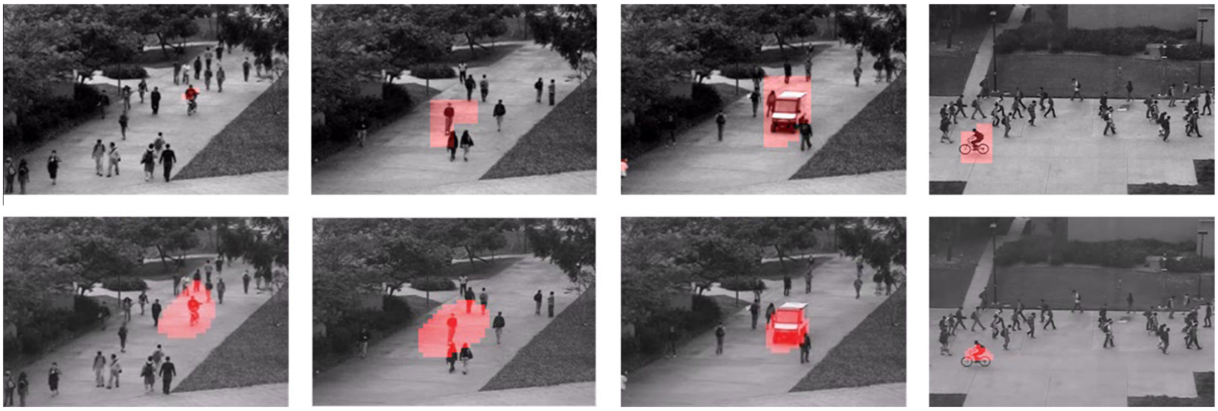

Fig. 10. Anomaly localization results (top) compared with the best performing method [21] (bottom) on the UCSD dataset.

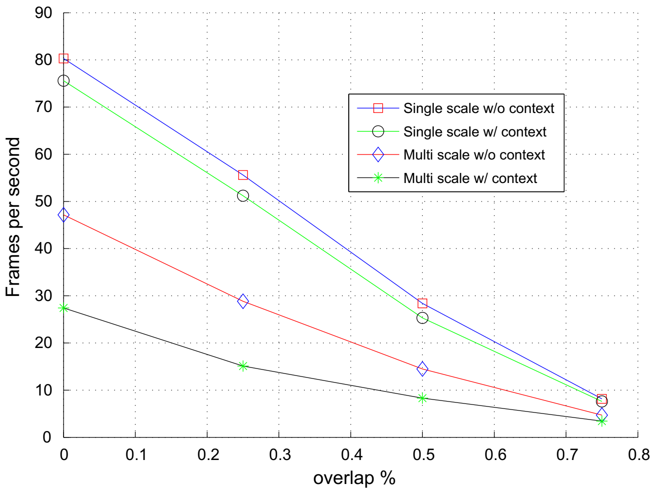

Fig. 11. Comparison of the number of frames per second (FPS) processed while varying the spatial overlap of cuboids, using single-scale and multi-scale approach on the Peds1 dataset.

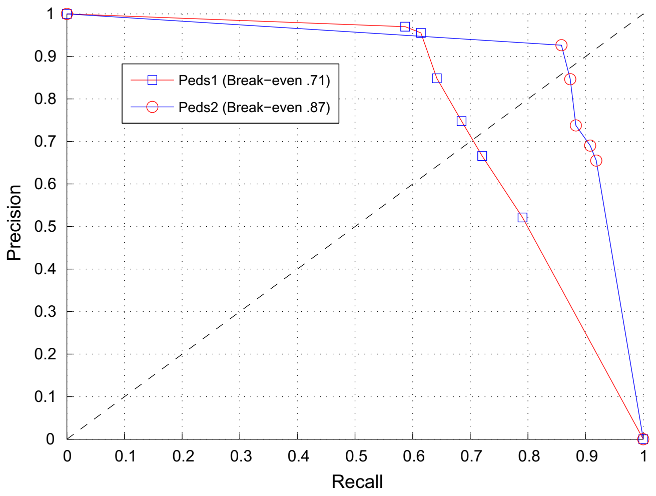

Fig. 12. Precision/recall curve of our approach for the two datasets. Break-even, i.e. the intersection with the dashed line, is reported in the legend.

only 3–6% of the total computation time while using the context it increases to 11–12% of the total computation time. For the multi-scale approach, the use of smaller patches (10 × 10) increases the burden of context modeling. Even with multiple scales and contextual neighborhood queries our system is able to process 8 frames per second, with 50% patch overlap, and to obtain competitive results of detection and localization with respect to non-realtime systems that require several seconds to process a single frame [34]. We expect that code optimization exploiting modern multi-core CPUs will greatly reduce the computational gap between multi-scale and single-scale methods. Cuboid size does not affect the computation time since smaller cuboids imply an increased number of descriptors which are faster to compute while bigger cuboids generate fewer but slower to compute descriptors. The main reason for the decrease of computational performance when using the multi-scale approach is the increased number of model queries made when using smaller cuboids.

Since in video surveillance the precision of the alarms is important, because a human operator may be disturbed by a high number of false alarms, in Fig. 12 we report the precision-recall curve for the UCSD dataset, created varying the p_a parameter from 10^{-5} to 10^{-2}, showing a good performance; considering low probabilities p_a for the anomalies the recall is reduced while raising the precision, and *vice versa*. In particular, the break-even point a 0.71 of precision and recall is obtained for $10^{-4} \leqslant p_a \leqslant 10^{-3}$ for Peds1 while for Peds2 the value of .87 is obtained for $10^{-5} \leqslant p_a \leqslant 10^{-4}$.

5. Conclusions

In this paper we have presented a multi-scale non-parametric anomaly detection approach that can be executed in real-time in a completely unsupervised manner. The approach is capable of localizing anomalies in space and time. We have also provided a straightforward procedure to dynamically update the learned model, to deal with scene changes that happen in real-world surveillance scenarios. Dense and overlapping spatio-temporal features, that model appearance and motion information, have been used to capture the scene dynamics, allowing the detection of different types of anomalies. The proposed method is capable of handling challenging crowded scenes that cannot be modeled using trajectories or pure motion statistics (optical flow).

A comparison on a publicly available dataset shows that our method achieves the best performance with respect to existing state-of-the-art real-time solutions [19,21,22].

References

[1] L. Seidenari, M. Bertini, A. Del Bimbo, Dense spatio-temporal features for non-parametric anomaly detection and localization, in: Proc. of ACM Workshop on Analysis and Retrieval of Tracked Events and Motion in Imagery Streams (ARTEMIS), 2010, pp. 27–32.
[2] T. Troscianko, A. Holmes, J. Stillman, M. Mirmehdi, D. Wright, A. Wilson, What happens next? The predictability of natural behaviour viewed through CCTV cameras, Perception 33 (1) (2004) 87–101.
[3] H. Keval, M. Sasse, "Not the usual suspects": a study of factors reducing the effectiveness of CCTV, Secur. J. 23 (2) (2010) 134–154.
[4] N. Haering, P. Venetianer, A. Lipton, The evolution of video surveillance: an overview, Mach. Vis. Appl. 19 (5–6) (2008) 279–290.
[5] A. Stedmon, S. Harris, J. Wilson, Simulated multiplexed CCTV: The effects of screen layout and task complexity on user performance and strategies, Secur. J. 24 (2011) 344–356.
[6] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: a survey, ACM Comput. Surv. 41 (3) (2009) 15:1–15:58.
[7] C. Brax, L. Niklasson, M. Smedberg, Finding behavioural anomalies in public areas using video surveillance data, in: Proc. of 11th International Conference on Information Fusion, 2008.
[8] I. Ivanov, F. Dufaux, T.M. Ha, T. Ebrahimi, Towards generic detection of unusual events in video surveillance, in: Proc. of IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2009.
[9] P. Antonakaki, D. Kosmopoulos, S.J. Perantonis, Detecting abnormal human behaviour using multiple cameras, Signal Process. 89 (9) (2009) 1723–1738.
[10] S. Calderara, C. Alaimo, A. Prati, R. Cucchiara, A real-time system for abnormal path detection, in: Proc. of 3rd IEE International Conference on Imaging for Crime Detection and Prevention (ICDP 2009), London, UK, 2009.
[11] J. Li, S. Gong, T. Xiang, Discovering multi-camera behaviour correlations for on-the-fly global activity prediction and anomaly detection, in: Proc. of IEEE International Conference on Computer Vision Workshops (ICCV Workshops), 2009, pp. 1330–1337.
[12] C. Liu, G. Wang, W. Ning, X. Lin, L. Li, Z. Liu, Anomaly detection in surveillance video using motion direction statistics, in: Proc. of IEEE International Conference on Image Processing (ICIP), 2010, pp. 717–720.
[13] C. Piciarelli, G.L. Foresti, Surveillance-oriented event detection in video streams, IEEE Intelligent Systems 26 (3) (2011) 32–41.
[14] C.C. Loy, T. Xiang, S. Gong, Detecting and discriminating behavioural anomalies, Pattern Recogn. 44 (1) (2011) 117–132.
[15] D. Zhang, D. Gatica-Perez, S. Bengio, I. McCowan, Semi-supervised adapted HMMs for unusual event detection, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2005, pp. 611–618.
[16] R. Sillito, R. Fisher, Semi-supervised learning for anomalous trajectory detection, in: Proc. of British Machine Vision Conference (BMVC), Citeseer, 2008, pp. 1035–1044.
[17] C. Piciarelli, G. Foresti, On-line trajectory clustering for anomalous events detection, Pattern Recogn. Lett. 27 (15) (2006) 1835–1842 (Vision for Crime Detection and Prevention).
[18] O. Boiman, M. Irani, Detecting irregularities in images and in video, Int. J. Comput. Vision (IJCV) 74 (1) (2007) 17–31.
[19] A. Adam, E. Rivlin, I. Shimshoni, D. Reinitz, Robust real-time unusual event detection using multiple fixed-location monitors, IEEE Trans. Pattern Anal. Mach. Intell. (PAMI) 30 (3) (2008) 555–560.

[20] T. Xiang, S. Gong, Video behavior profiling for anomaly detection, IEEE Trans. Pattern Anal. Mach. Intell. (PAMI) 30 (5) (2008) 893–908.

[21] R. Mehran, A. Oyama, M. Shah, Abnormal crowd behavior detection using social force model, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[22] J. Kim, K. Grauman, Observe locally, infer globally: a space–time MRF for detecting abnormal activities with incremental updates, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[23] F. Jiang, Y. Wu, A. Katsaggelos, A dynamic hierarchical clustering method for trajectory-based unusual video event detection, IEEE Trans. Image Process. (TIP) 18 (4) (2009) 907–913.

[24] M. Breitenstein, H. Grabner, L. Van Gool, Hunting Nessie: Real time abnormality detection from webcams, in: Proc. of ICCV International Workshop on Visual Surveillance, 2009.

[25] F. Jiang, Y. Wu, A. Katsaggelos, Detecting contextual anomalies of crowd motion in surveillance video, in: Proc. of IEEE International Conference on Image Processing (ICIP), 2009, pp. 1117–1120.

[26] J. Yin, Y. Meng, Abnormal behavior recognition using self-adaptive hidden markov models, in: M. Kamel, A. Campilho (Eds.), Image Analysis and Recognition, Lecture Notes in Computer Science, vol. 5627, Springer, Berlin/Heidelberg, 2009, pp. 337–346.

[27] J. Varadarajan, J.-M. Odobez, Topic models for scene analysis and abnormality detection, in: Proc. of IEEE International Conference on Computer Vision Workshops (ICCV Workshops), 2009, pp. 1338–1345.

[28] F. Jiang, J. Yuan, S.A. Tsaftaris, A.K. Katsaggelos, Anomalous video event detection using spatiotemporal context, Comput. Vision Image Understand. (CVIU) 115 (3) (2011) 323–333 (Special Issue on Feature-Oriented Image and Video Computing for Extracting Contexts and Semantics).

[29] L. Kratz, K. Nishino, Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009, pp. 1446–1453.

[30] S. Khalid, Activity classification and anomaly detection using m-medoids based modelling of motion patterns, Pattern Recogn. 43 (10) (2010) 3636–3647.

[31] R. Castellanos, H. Kalva, O. Marques, B. Furht, Event detection in video using motion analysis, in: Proc. of ACM Workshop on Analysis and Retrieval of Tracked Events and Motion in Imagery Streams (ARTEMIS), 2010, pp. 57–62.

[32] I. Laptev, On space–time interest points, Int. J. Comput. Vision (IJCV) 64 (2–3) (2005) 107–123.

[33] P. Dollar, V. Rabaud, G. Cottrell, S. Belongie, Behavior recognition via sparse spatio-temporal features, in: Proc. of VSPETS, 2005.

[34] V. Mahadevan, W. Li, V. Bhalodia, N. Vasconcelos, Anomaly detection in crowded scenes, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 2010.

[35] F. Jurie, B. Triggs, Creating efficient codebooks for visual recognition, in: Proc. of IEEE International Conference on Computer Vision (ICCV), 2005.

[36] H. Wang, M.M. Ullah, A. Kläser, I. Laptev, C. Schmid, Evaluation of local spatio-temporal features for action recognition, in: Proc. of British Machine Vision Conference (BMVC), 2009, p. 127.

[37] L. Ballan, M. Bertini, A. Del Bimbo, L. Seidenari, G. Serra, Recognizing human actions by fusing spatio-temporal appearance and motion descriptors, in: Proc. of IEEE International Conference on Image Processing (ICIP), Cairo, Egypt, 2009.

[38] J.R.R. Uijlings, A.W.M. Smeulders, R.J.H. Scha, Real-time bag of words, approximately, Proceeding of the ACM International Conference on Image and Video Retrieval, CIVR '09, ACM, New York, NY, USA, 2009, pp. 6:1–6:8. doi:http://doi.acm.org/10.1145/1646396.1646405.

[39] P. Scovanner, S. Ali, M. Shah, A 3-dimensional SIFT descriptor and its application to action recognition, in: Proc. of ACM Multimedia, 2007.

[40] L. Ballan, M. Bertini, A. Del Bimbo, L. Seidenari, G. Serra, Effective codebooks for human action categorization, in: Proc. of ICCV International Workshop on Video-oriented Object and Event Classification (VOEC), Kyoto, Japan, 2009.

[41] W.T. Freeman, E.C. Pasztor, Learning low-level vision, Int. J. Comput. Vision (IJCV) 40 (1) (2000) 25–47.

[42] M. Muja, D.G. Lowe, Fast approximate nearest neighbors with automatic algorithm configuration, in: Proc. of International Conference on Computer Vision Theory and Application, INSTICC Press, 2009, pp. 331–340.