# A Proposal for Local and Global Human Activities Identification

Antonio Fernández-Caballero,
José Carlos Castillo, and José María Rodríguez-Sánchez

Departamento de Sistemas Informáticos & Instituto de Investigación en Informática
Universidad de Castilla-La Mancha, Campus Universitario s/n, 02071-Albacete, Spain
caballer@dsi.uclm.es

**Abstract.** There are a number of solutions to automate the monotonous task of looking at a monitor to find suspicious behaviors in video surveillance scenarios. Detecting strange objects and intruders, or tracking people and objects, is essential for surveillance and safety in crowded environments. The present work deals with the idea of jointly modeling simple and complex behaviors to report local and global human activities in natural scenes. In order to validate our proposal we have performed some tests with some CAVIAR test cases. In this paper we show some relevant results for some study cases related to visual surveillance, namely "speed detection", "position and direction analysis", and "possible cash-point holdup detection".

**Keywords:** Human activities, simple behaviors, complex behaviors.

## 1 Introduction

Detecting strange objects and intruders, or tracking people and objects, is essential for surveillance and safety in crowded environments [21], [9]. Much research has been dedicated to understanding human activities in the last decade (e.g. [10], [4]). Advanced visual surveillance systems not only need to track moving objects but also interpret their patterns of behavior [5]. Generally, these systems can detect a few simple concepts in video streams. The task of activity recognition is to bridge the gap between numerical pixel level data and a high-level abstract activity description. Activities analysis consists of feature extraction, basic activity description and complex activity description. Complex activities are composed of many single activities with their temporal relations. According to the features used for analysis, the activity analysis methods can be classified into three kinds, spatial based (such as shape), motion based (such as trajectory), and spatial-temporal based methods. Many techniques and methods have been used so far in human activity recognition and understanding. According to [12], shape features and spatial-temporal features are often used for single person activity analysis, and motion features can be used for interactive person activity.

F.J. Perales and R.B. Fisher (Eds.): AMDO 2010, LNCS 6169, pp. 78–87, 2010.

Bayesian networks have been used to recognize static postures or simple events. In [13] an activity recognition approach is proposed in which an activity is decomposed into multiple interactive stochastic processes, each corresponding to one scale of motion details. In [16] abnormal activities involving two persons using Recurrent Bayesian networks (RBNs) are detected. Recently, in [29] a novel unsupervised learning framework to model activities and interactions in crowded and complicated scenes is proposed. Inspired by the applications in speech recognition, the hidden Markov model (HMM) formalism has been extensively applied to activity recognition (e.g. [8]). In [2] an automatic technique is proposed for detection of abnormal events in crowds where the motion models are HMMs to cope with the variable number of motion samples that might be present in each observation window. In [25] a Bayesian computer vision system for modeling and recognizing human interactions using CHMMs and HMMs is described. Another approach [11] models scenario events from shape and trajectory features using a hierarchical activity representation, where events are organized into several layers of abstraction, providing flexibility and modularity in modeling scheme. In [1] a real-time system to detect context-independent events in video shots is proposed. In [15], recent approaches of video event understanding are presented. The importance of the two main component of the event understanding process – abstraction and event modeling– is also pointed out. Abstraction corresponds to the process of molding the data into informative units to be used as input to the event model [26,14,27,22] while event modeling is devoted to describing events of interest formally and enabling recognition of these events as they occur in the video sequence [28]. Our approach is closely related to the works of Ivanov and Bobick [13] and Hongeng et al. [11] in the sense that the external knowledge about the problem domain is incorporated into the expected structure of the activity model. Motion-based image features are linked explicitly to a symbolic notion of hierarchical activity through several layers of more abstract activity descriptions. Atomic actions are detected at a low level and fed to hand-crafted grammars to detect activity patterns of interest. Our inspiration also is close to the paper by [1], as we work with shape and trajectory to indicate the events related to moving objects.

## 2 Description of Local and Global Activities

Analyzing a video scene entails two large phases. On the one hand, we have the first phase in object detection [19], namely segmentation (e.g. [6], [20], [18]) and tracking o (e.g. [7], [17]). This phase consists of capturing images, analyzing them for shape interpretation and afterwards, recognizing them throughout the scene. On the other hand, we have the part this work focuses on, namely, scene interpretation (context recognition), made up of *basic actions* interpretation, *global behavior* interpretation and finally, interpretation of the *scene on a global scale*. In our proposal, the purpose of activity description is to reasonably choose a group of motion words or shout expressions to report activities of moving objects or humans in natural scenes.

### 2.1 Objects of Interest

From the ETISEO (see `http://www-sop.inria.fr/orion/ETISEO/index.htm`) classification, four categories are established for dynamic objects and two for static objects. As for the first, we distinguish between a *person*, a *group of people* (made up of two or more people), a *portable object* (such as a brief case) and other dynamic objects (able to move on its own), classified as *moving object*. As for static objects, we will distinguish between *areas* and *pieces of equipment*. The latter can be labeled as a portable object if a dynamic object, people or group, interacts with it and it starts moving.

### 2.2 Description of Local Activities

In order to generalize the detection process we start with small functionalities which detect simple actions of the active objects in the scene. Using these functions, we build behavior patterns much more complex and suited for the aims of each video surveillance system. These small actions are defined by action indicative queries about the actions performed by an active object (see Table 1).

**Table 1.** Local activities

| Action | Origin vertex | Destination vertex |
|---|---|---|
| Object-like | Object speed | Makes it possible to define if an object is still, walking, running, going at great speeds, etc. |
| | Object trajectory | Apart from speed, we can obtain the direction and moving direction of an object. |
| Environment interaction | Direction | The system must determine if a person is approaching a specific area of the scenario. By taking the object's speed and trajectory as reference, the object's ultimate goal is inferred. |
| | Position | By knowing the important areas of the scenario, the system is capable of determining the relative position of dynamic objects. This way, it can detect if a person is standing in one of the areas. |
| Object interaction | Proximity | The system must detect the distance between objects. |
| | Orientation | The system determines whether an object is approaching another or whether they are both approaching each other. |
| | Grouping | The system uses the parameters generated in the two previous points to detect object grouping (by taking into account its proximity and direction). |

### 2.3 Description of Global Activities

Interpreting a visual scene is a task which, in general, resorts to a large body of prior knowledge and experience of the viewer [23]. Through the actions or queries described in the previous section, we can find out basic patterns (an object speed or direction) and more complex patterns (e.g. the theft of a purse). It is essential to define the desired behavior pattern in each situation, by using the basic actions or queries from the previous section. For each specific scene, a state diagram and a set of rules are designed to indicate the patterns.

The proposed video surveillance system will be able to detect simple actions or queries and adapt to a great deal of situations. Also, it will be configured to detect the behavior patterns necessary in each case and associate an alarm level to each one which will enable them to be filtered and have a priority associated.

## 3 Image Preprocessing

Input image segmentation is not enough to detect the activities in the scene, other data which are not included (speed and direction) are necessary. Thus, the system takes the initial segmentation data and infers the new necessary parameters. For it, the preprocessing techniques described in Table 2 are necessary.

**Table 2.** Preprocessing techniques

| Preprocessing | Details |
|---|---|
| **Speed Hypothesis** | The average speed for each object is calculated by dividing the displacement ($\Delta x$) by the time that has elapsed ($\Delta t$) in each frame. |
| **Direction and Moving Direction Hypothesis** | To find out the direction of objects, we calculate the angle of the straight line that passes through the positions of the previous and current instants in each object. |
| **Image Rectification** | Perspective distortion occurs because the distance between the furthest points from the camera is less than the distance between the closest points. The real position is measured through the weighted distance measure of the four manually placed points closest to the position we wish to interpolate. |
| **Data Smoothing** | The data taken at two time instants will be separated with enough time to avoid small distortions but this distance will be small enough to enable accurate results. We will call this distance between both consecutive time instants, interval analysis. At each interval analysis, the value of the hypotheses is updated, but the old value is not automatically substituted for the new one. To calculate the value at that instant, we calculate the means for both values. |

## 4 Specification of Behaviors

### 4.1 Specification of Simple Behaviors

The system should be able to respond to a series of queries intended to find out behavior patterns of objects in the scene (see Table 3). These queries are defined as functions and return a logical value, which will be true if they are fulfilled for a specific object. They are represented in the following format:

$$query\ (parameter_1, parameter_2, ..., parameter_n)$$

### 4.2 Specification of Complex Behaviors

Patterns at a global level are used to analyze the scene from a general point of view without focusing on any specific object (detect patterns where more than one object intervenes).

**Local Complex Behaviors.** Objects in the scene are associated to a state machine that indicates the state they are in (what they are doing at that time instant). This state machine can be seen as a directed graph where the vertices are the possible states of the object and the edges are the basic functions or queries previously discussed. An edge has at least one associated outcome of the assessment (true or false) of a query, indicating an action of object, query $q_i$.

**Table 3.** Simple Queries

| Type | Query | Description |
|---|---|---|
| Movement-based | *hasSpeedBetween* (*min*, *max*) | It is fulfilled if the object moves at a speed within the range $[min, max]$. |
| | *hasSpeedGreaterThan* (*speed*) | It is fulfilled if the object moves at a speed greater than that indicated in the parameter *speed*. |
| Orientation-based Direction | *hasDirection* (*staticObject*) | It is fulfilled if the object is headed towards *staticObject*, being *staticObject* a static object in the scene. |
| | *isFollowing* () | It is true if a dynamic object is following a non-dynamic object. We use the displacement angle. |
| Location-based | *isInsideZone* (*staticObject*) | It is true if a dynamic object is on the static object *staticObject*. |
| | *isCloseTo* (*distance*, *staticObject*) | It is fulfilled if the object is closer than distance from the static object *staticObject*. |
| | *enterInScene* () | It is fulfilled when the object appears in the scene for the first time. |

Therefore, an edge can have more than one query associated to it. For an edge with several actions to be fulfilled, all the associated queries have to be fulfilled. If a more complex rule is needed, where disjunctions also appear so that an object changes states, the rule must be divided into two edges.

**Global Complex Behaviors.** To detect global behavior patterns, more than just the local state machine from the previous section is needed since only the state of each object in that machine is reflected separately. These patterns are represented through state machines which vertices represent a possible state in the scene. Just like in the local state machine, the edges are made up of a series of queries that must be fulfilled at a certain time for the scene to change states.

## 5 Data and Results

In order to validate our proposal we have opted for working with the test cases that CAVIAR (coming from the EC Funded CAVIAR project/IST 2001 37540, found at URL: `http://homepages.inf.ed.ac.uk/rbf/CAVIAR/`) makes available for researchers. In fact, the test cases offer ground truth data; this enables bypassing the segmentation phase and only focusing on the problem of human activities identification. Of course, due to the limitation in pages of the current article, only a very limited set of examples may be provided. Concretely, in this paper we show some relevant results for the following study cases: "speed detection", "position and direction analysis", and "possible cashpoint holdup detection". This is a usual approach (e.g. [3], where the detection and classification of fighting and pre and post fighting events when viewed from a video camera is investigated).

### 5.1 Image Preprocessing

We select the first frame in any scene as backdrop image to make the placement of control points and fixed objects easier. Control points are used to compensate

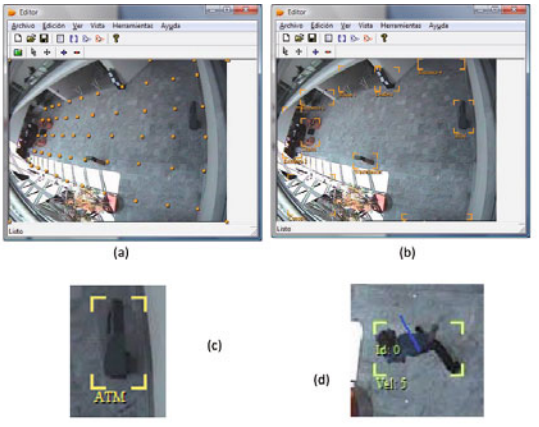


**Fig. 1.** Test environment for CAVIAR. (a) Position point maps. (b) Fixed objects in the scene. (c) Labeling of a static object in the scene. (d) Labeling of a dynamic object in the scene.

image distortion caused by the perspective and camera lens. Control points are interpolated using the four reference positions provided by CAVIAR (see Fig 1a). After creating the point map, we point out the fixed objects in the scene that will interact with the dynamic objects (as shown in Fig. 1b). In Fig. 1c and 1d, you may find examples of the labeling of a static object and a dynamic object, respectively, in the scene.

### 5.2 Speed Detection

In this case, we detect if the person starts running or moves slowly. To do this, we use the queries "hasSpeedBetween" and "hasSpeedGreaterThan" with their associated local state diagram (see Fig. 2a). When adjusting the alarm level to *I* and analyzing scene `Browse2` (series *Browsing*, case *Person browsing and reading for a while*), we get the output shown in Table 4. If the alarm level is adjusted to *II* and scene `Fight_RunAway1` (series *Two people fighting*, test case *Two people meet, fight and run away*) is analyzed, we get Table 5 as output. As shown, the application has detected the time when the two people started running.

### 5.3 Position and Direction Analysis

A configuration was designed for the purpose not only to analyze the position of people in a scene, but also to predict if someone is headed towards a specific position. Queries "isInsideZone" and "isCloseTo" are used to detect position and query "hasDirection" in order to generate a direction hypothesis. Fig. 2b shows the automaton that detects if someone is headed towards or is at the wastebasket, the leaflets, the seats or the cashpoint. Tests on three different scenes have been run. Table 6 shows the results from the analysis of scenes `Rest_InChair`, `Browse2`, and `Browse3`.

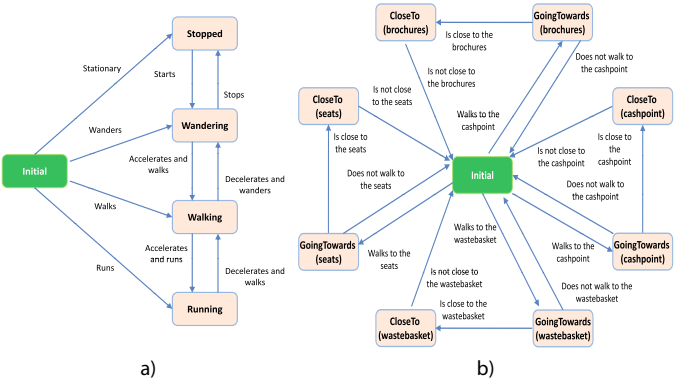


**Fig. 2.** Local diagrams. (a) Speed detection. (b) Position and direction analysis.

There are false positives in the last two tests. They are in the 14th second of test case `Browse2` and in the 15th second of test case `Browse3`. Indeed, object 3 was not going to the wastebasket but the direction of the object at that time made it seem like it could be going there. To avoid this, we could add another rule to edge direction to avoid predicting a possible target if the object is too far away. We could add an "isCloseTo" query to act along with queries "hasDirection" and "hasSpeedGreaterThan".

**Table 4.** Results of speed detection in scene "Browse2"

| Time | Object | State | Alarm | Time | Object | State | Alarm |
|---|---|---|---|---|---|---|---|
| 0:00:00 | 0 | Stopped | I | 0:00:12 | 3 | Walking | I |
| | 1 | Stopped | | | 3 | Wandering | |
| | 1 | Wandering | | 0:00:13 | 3 | Walking | I |
| 0:00:01 | 1 | Walking | I | 0:00:14 | 3 | Wandering | I |
| 0:00:04 | 1 | Wandering | I | 0:00:15 | 3 | Walking | I |
| 0:00:05 | 1 | Stopped | I | 0:00:21 | 3 | Wandering | I |
| | 1 | Wandering | | 0:00:22 | 3 | Stopped | I |
| 0:00:06 | 1 | Walking | I | 0:00:30 | 3 | Wandering | I |
| 0:00:07 | 2 | Walking | I | | 3 | Walking | |
| | 2 | Wandering | | 0:00:33 | 3 | Wandering | I |
| | 2 | Stopped | | | 3 | Stopped | |
| 0:00:09 | 1 | Wandering | I | | | | |
| | 2 | Wandering | | | | | |

## 5.4 Possible Cashpoint Holdup Detection

It is also possible to design configurations able to detect suspicious behaviors. Here is an example pertaining to a situation related to a cashpoint. First, a local state diagram is created to detect the different ways of getting to the cashpoint. With this graph, we will be able to know if someone is going to the cashpoint, how fast he/she is going and if he/she is already next to the cashpoint. In the local state diagram shown in Fig. 3a, we see how a person can go into three states from the initial state: going towards the cashpoint slowly, walking or running.

**Table 5.** Results of speed detection in scene "Fight_RunAway1"

| Time | Object | State | Alarm |
|---|---|---|---|
| 0:00:15 | 7 | Running | II |
| 0:00:16 | 6 | Running | II |

**Table 6.** Results of position and direction analysis in scenes "Rest_InChair", "Browse2" and "Browse3"

| Scene | Time | Object | State | Alarm |
|---|---|---|---|---|
| Rest_InChair | 0:00:13 | 1 | GoingTowards (seats) | II |
| | 0:00:16 | 1 | InsideZone (seats) | III |
| Browse 2 | 0:00:14 | 3 | GoingTowards (wastebasket) | II |
| | 0:00:16 | 3 | GoingTowards (cashpoint) | II |
| | 0:00:21 | 3 | CloseTo (cashpoint) | III |
| Browse 3 | 0:00:15 | 3 | GoingTowards (cashpoint) | II |
| | 0:00:20 | 3 | GoingTowards (leaflets) | II |
| | 0:00:20 | 3 | CloseTo (leaflets) | III |

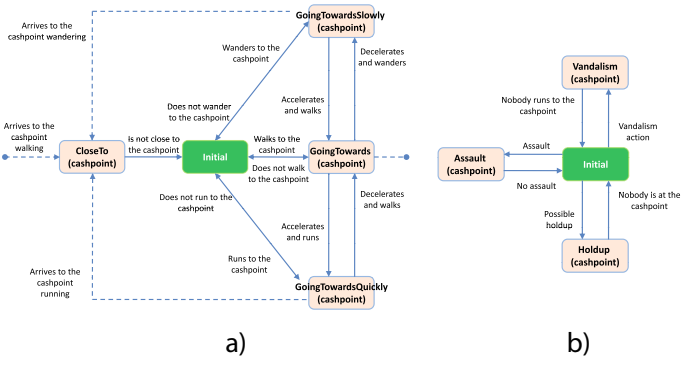


**Fig. 3.** Holdup at a cashpoint. (a) Local diagram. (b) Global diagram.

Thus two parameters are controlled, a person's speed and whether or not a person is going to the cashpoint.

Once the local state diagram has been created, we go on to behavior pattern specification at global level in the scene. Fig. 3b shows how the automaton is able to detect the suspicious behaviors described. Indeed, the diagram of Fig. 3b can detect suspicious behaviors, such as when there is someone at the cashpoint and someone else approaches him/her slowly. It can also detect if there is someone at the cashpoint and one or more people run towards him/her. Lastly, it can detect possible vandalism at the cashpoint. It will detect if one or more people run to the cashpoint and there is no one using it.

## 6 Conclusions

In this paper, an approach to human activities detection in complex scenarios has been presented. The approach describes two levels in which activities should

be considered: local activities are necessary to generalize the detection process; and global activities are used to detect behavior patterns that involve not only a single object, but also groups of objects (or even the whole set of objects) in the scene. Some parameters must be inferred from the objects in the scene, such as speed or direction. The system takes the initial segmentation to calculate these parameters. Next, a set of queries are proposed in order to specify simple behaviors (to detect movement, orientation and location of the objects), and complex behaviors (where one or several objects intervenes).

In comparison to other approaches, such as Bayesian Networks or HMMs [24], our proposal is not able to model uncertainty in video events; but it is presented as a useful tool in video event understanding because of its simplicity, its ability to model temporal sequence and its ability to easily incorporate new actions. The results obtained so far are promising and we are currently engaged in performing test with real segmented data taken from different scenarios.

## Acknowledgements

This work was partially supported by the Spanish Ministerio de Ciencia e Innovación under projects TIN2007-67586-C02-02 and TIN2010-20845-C03-01, and by the Spanish Junta de Comunidades de Castilla-La Mancha under projects PII2I09-0069-0994 and PEII09-0054-9581.

## References

1. Amer, A., Dubois, E., Mitiche, A.: Rule-based real-time detection of context-independent events in video shots. Real-Time Imaging 11(33), 244–256 (2005)
2. Andrade, E.L., Blunsden, S., Fisher, R.B.: Modelling crowd scenes for event detection. In: 18th International Conference on Pattern Recognition, vol. 1, pp. 175–178 (2006)
3. Blunsden, S., Fisher, R.B.: Pre-fight detection - Classification of fighting situations using hierarchical AdaBoost. In: Fourth International Conference on Computer Vision Theory and Applications, vol. 2, pp. 303–308 (2009)
4. Bobick, A.F., Davis, J.W.: The recognition of human movement using temporal templates. IEEE PAMI 23(3), 257–267 (2001)
5. Buxton, H., Gong, S.: Visual surveillance in a dynamic and uncertain world. Artificial Intelligence 78(1), 431–459 (1995)
6. Fernández-Caballero, A., López, M.T., Saiz-Valverde, S.: Dynamic Stereoscopic Selective Visual Attention (DSSVA): Integrating motion and shape with depth in video segmentation. ESWA 34(2), 1394–1402 (2008)
7. Fernández-Caballero, A., Gómez, F.J., López-López, J.: Road-traffic monitoring by knowledge-driven static and dynamic image analysis. ESWA 35(3), 701–719 (2008)
8. Galata, A., Johnson, N., Hogg, D.: Learning variable-length Markov models of behavior. CVIU 81(3), 398–413 (2001)
9. Gascueña, J.M., Fernández-Caballero, A.: On the use of agent technology in intelligent, multi-sensory and distributed surveillance. In: KER (2009) (in press)
10. Gavrila, D.M.: The visual analysis of human movement: a survey. CVIU 73(1), 82–98 (1999)

11. Hongeng, S., Nevatia, R., Bremond, F.: Video-based event recognition: activity representation and probabilistic recognition methods. CVIU 96(2), 129–162 (2004)
12. Huang, K., Wang, S., Tan, T., Maybank, S.: Human behavior analysis based on a new motion descriptor. IEEE CirSysVideo 19(12), 1830–1840 (2009)
13. Ivanov, Y.A., Bobick, A.F.: Recognition of visual activities and interactions by stochastic parsing. IEEE PAMI 22(8), 852–872 (2000)
14. Laptev, I., Pérez, P.: Retrieving actions in movies. In: International Conference on Computer Vision, pp. 1–8 (2007)
15. Lavee, G., Rivlin, E., Rudzsky, M.: Understanding video events: a survey of methods for automatic interpretation of semantic occurrences in video. IEEE SMC-C 39(5), 489–504 (2009)
16. Loccoz, N.M., Bremond, F., Thonnat, M.: Recurrent Bayesian network for the recognition of human behaviors from video. In: 3rd International Conference on Computer Vision Systems, pp. 68–77 (2003)
17. López, M.T., Fernández-Caballero, A., Fernández, M.A., Mira, J., Delgado, A.E.: Dynamic visual attention model in image sequences. IMAVIS 25(5), 597–613 (2007)
18. López, M.T., Fernández-Caballero, A., Fernández, M.A., Mira, J., Delgado, A.E.: Motion features to enhance scene segmentation in active visual attention. Pattern Recognition Letters 27(5), 469–478 (2006)
19. López, M.T., Fernández-Caballero, A., Mira, J., Delgado, A.E., Fernández, M.A.: Algorithmic lateral inhibition method in dynamic and selective visual attention task: Application to moving objects detection and labelling. ESWA 31(3), 570–594 (2006)
20. López-Valles, J.M., Fernández, M.A., Fernández-Caballero, A.: Stereovision depth analysis by two-dimensional motion charge memories. Pattern Recognition Letters 28(1), 20–30 (2007)
21. Moreno-Garcia, J., Rodriguez-Benitez, L., Fernández-Caballero, A., López, M.T.: Video sequence motion tracking by fuzzification techniques. ASOC 10(1), 318–331 (2010)
22. Natarajan, P., Nevatia, R.: View and scale invariant action recognition using multiview shape-flow models. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
23. Neumann, B., Möller, R.: On scene interpretation with description logics. IMAVIS 26(1), 82–101 (2008)
24. Oliver, N.M., Horvitz, E.: A comparison of HMMs and dynamic Bayesian networks for recognizing office activities. User Modeling, 199-209 (2005)
25. Oliver, N.M., Rosario, B., Pentland, A.P.: A Bayesian computer system for modeling human interactions. IEEE PAMI 22(8), 831–843 (2000)
26. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
27. Tran, S.D., Davis, L.S.: Event modeling and recognition using Markov logic networks. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 610–623. Springer, Heidelberg (2008)
28. Ulusoy, I., Bishop, C.M.: Generative versus discriminative methods for object recognition. In: The 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 258–265 (2005)
29. Wang, X., Ma, X., Grimson, W.E.L.: Unsupervised activity perception in crowded and complicated scenes using hierarchical Bayesian models. IEEE PAMI 31(3), 539–555 (2009)