Towards a Multi-purpose Monocular Vision-based High-Level Situation Awareness System

David Münch, Kai Jüngling, and Michael Arens

Fraunhofer IOSB, Ettlingen, Germany,
{david.muench|kai.juengling|michael.arens}@iosb.fraunhofer.de

Abstract. In surveillance applications human operators are either confronted with a high cognitive load or monotonic time periods where the operator's attention rapidly decreases. Therefore, automatic high-level interpretation of image sequences gains increasing importance in assisting human operators. We present a generic hierarchical system that generates high-level logic-based situation descriptions in various domains. The system consists of two components. First, a vision component provides 3D spatial and temporal information about objects in scenes. Second, the situation recognition component uses knowledge encoded in Situation Graph Trees and a fuzzy graph traversal allowing exhaustive situation awareness. The system is tested with real video data comprising persons, their actions, and interactions. In order to show the domain independence we used recorded data from moving vehicles and static surveillance cameras. The results show that the system is usable with multi modal data and can easily be modified and extended.

Keywords: situation recognition, ISM tracking, Fuzzy Metric Temporal Logic, Situation Graph Tree

1 Introduction

A computer vision-based situation awareness system is a challenging task with an enormous amount of potential applications. One could be a distinct area to be observed by an operator for threatening situations. The situation awareness system does the observation automatically and only alarms the operator if a predefined threatening situation has been recognized. Another application is a driver assistance system which observes the scene in front of the vehicle. The situations recognized can be used to inform the driver of possible threatening situations – either for the vehicle itself or some other vehicles or pedestrians. Numerous work has already been done in the field of person detection, person tracking and situation recognition.

A basis component of every vision system is detection and tracking of relevant objects in image sequences. With the progress made in the area of local image features in the past few years, approaches for object detection and tracking have evolved from simple foreground and motion detection algorithms [13, 25] to more sophisticated algorithms approaching the problem in a different way

(see [8] for an extensive survey). Rather than processing a detected foreground region as an object instance of a certain class, these latter approaches [6, 18, 28, 30] employ machine learning techniques to train object class specific models which are used to detect object class instances in input imagery. Many tracking approaches like [9, 16, 19] build on these dedicated object detectors to perform tracking in image sequences. Main advantages of these approaches over motion or foreground based approaches are that (i) different object classes can be distinguished unambiguously, (ii) they work despite camera motion and (iii) they are most widely stable against environmental conditions. Good performance of these approaches both for detection and tracking of multiple instances of a single object class has been shown in several papers [4, 16, 30].

The high-level interpretation of image sequences can be divided into three types of approaches. Recent surveys dealing with the interpretation of situations in image sequences are [17, 27]. First, there are statistical approaches using, e.g., Bayesian networks, hidden Markov models, or dynamic Bayesian networks. In general, the likelihood between the learned situation and the image sequence is determined by the models' probability of labeled situations (see [1, 5, 22]). Second, syntactic approaches apply nested production rules as used in formal grammars. This is followed by parsing the generated situation strings (see [12, 14]). Third, description-based approaches are built upon the formulation of temporal and spatial properties of situations and their hierarchical structure. In 1994 [2] introduced the interval temporal logic which allows advanced relations between intervals. Extended and improved by [29] the situation recognition became more efficient. Further improvements were made in [23] allowing the recognition of more complex situations. To deal with uncertainty in input data from sensors Tran and Davis [26] combined logical and probabilistic approaches. They use first order predicate logic with weighted rules as input for a Markov Logic Network. Another approach is the conceptual description of situations with Fuzzy Metric Temporal Logic (FMTL) and Situation Graph Trees (SGT) introduced by Nagel and his group [21]. In [3, 10] it is shown that FMTL/SGTs can be applied to traffic scenarios. Gonzàlez [11] applied FMTL/SGTs on situations with humans on a high level of abstraction.

All the approaches mentioned above have a hard coded fixed world model which does not allow the use of a moving camera. Our approach is a system combined of tracking and detection of objects, the estimation of 3D information about objects without a hard coded fixed world model, and the fuzzy recognition of high-level complex situations.

2 Vision Subsystem

To detect and track persons in image sequences we build on the work of Jüngling and Arens [16]. The key idea of this tracking approach is that the Implicit Shape Model (ISM) [18], which is a trainable object detection approach that builds on local features (we use SIFT [20] in this paper), is extended for tracking.

The ISM object detector and thus the tracking works on the basis of a codebook for a specific object category which is built in a training step based on

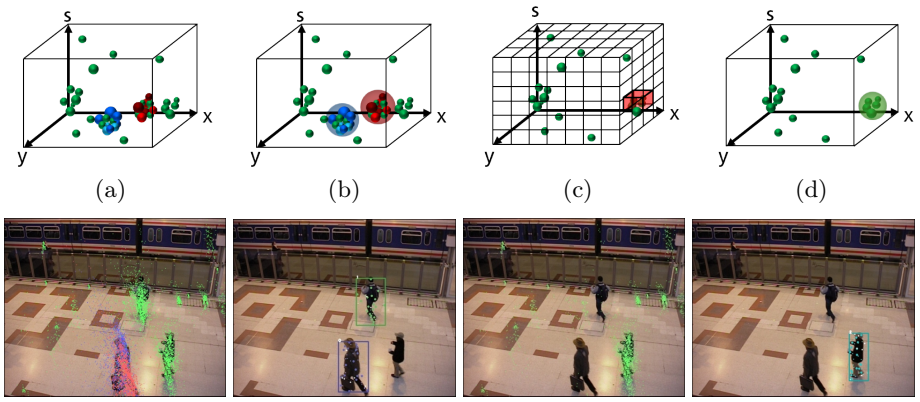
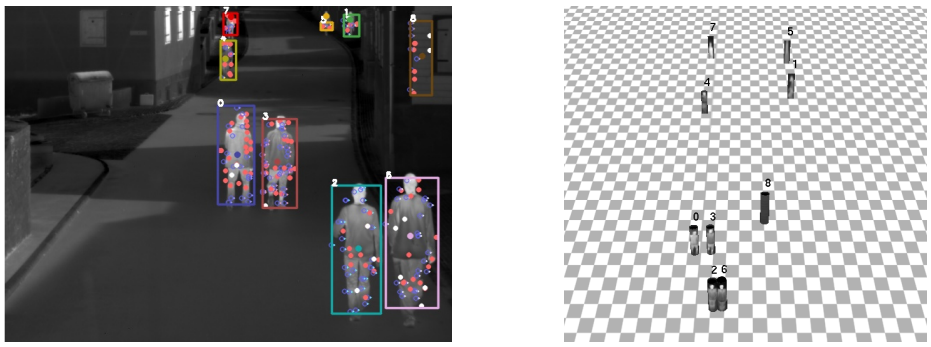


Fig. 1. Object tracking in the Hough-voting space (axes: x- and y-position and scale). (a) Joint voting space: Colors indicate vote type. Green votes do not have a correspondence to a former hypotheses feature and thus can vote for any object. Other colors have correspondences to former hypothesis features and vote only for specific objects hypotheses. (b) Mean-shift-search for tracking of known objects. Mean-shift search can be started for an existing hypothesis directly from the last known position of the hypothesis. No initial maxima search is necessary to identify possible object hypotheses since the starting points for the searches are already known. (c) Maxima search in the discretized Hough-space to detect new objects which are not known in the system yet. (d) Mean-shift search for refinement of maxima position of new hypotheses (see, too, [16]).

sample images of the object category of interest. Local image features extracted from the training samples on multiple scales are input to a clustering that identifies reoccurring features which are significant for the object category. Cluster centers build the prototypes for the initial codebook which describes the object category generically.

Tracking builds on this ISM detection and extends it by integrating temporal information into the Hough-based ISM object detection approach. This temporal extension for tracking is performed on the level of SIFT features which describe the object hypotheses. All hypotheses already known in the system at a time T (this can be an empty hypotheses set – new objects are detected and integrated as hypotheses automatically) are predicted for the current instant of time on the level of features to predict feature motion. The hypothesis features are then matched with SIFT features extracted from the current input image to build feature correspondences. These correspondences form the temporal information integration. This integrated information is then input to the standard object detection procedure, see Figure 1 and [16] for details.

Figure 1 shows how tracking is performed in this voting space: (a) shows the voting space. Here, green votes can vote for any object (also for new objects which have no hypothesis assigned at this instant of time), blue and red votes vote for the known object hypotheses with ID 1 and 2 respectively. As shown



(a) (b)

Fig. 2. Visualization of 3D estimation using 2D tracking data. (a) 2D tracking results in an urban scene with an infrared moving camera. (b) 3D bird's-eye view of the information of (a) after distance estimation.

in Figure 1 (b), tracking is performed by starting a mean-shift maximum search for every known hypothesis independently. This search only includes general votes (green) and votes for this specific hypothesis. After convergence of the mean-shift search, votes inside the mean-shift kernel are used to update the hypothesis. New objects which have not been seen before are detected in the 'reduced voting space' as shown in Figure 1 (c) and (d). Here, all votes which already contributed to a hypothesis and those which vote only for a 'known object' are removed. All remaining votes can form new object hypotheses. For that, the standard object detection is performed. New object hypotheses are transfered to the set of 'known hypotheses'.

The result of this tracking are image coordinate trajectories of all tracked instances of the object class of interest. At every instant of time, the system provides the positions of object hypotheses whose identities are determined by a unique object ID while they continuously appears in the camera's field of view (note that extensions for person reidentification have been proposed in [15]). In addition to ID and center position, object hypotheses are described by a set of SIFT features which also determine the bounding box extend.

Besides the main advantages of this tracking approach over other approaches, namely that it allows for tracking during short-term occlusion, tracking despite strong camera motion, object class and sensor modality independent tracking, it is able to estimate person scale. Scale estimation does not depend on the bounding box size, which would be sensitive to distortion and only work when the object is fully visible, but is automatically determined in object detection by the search in the 3D Hough-voting-space comprising x- and y-position and scale, and thus is based on semantic object class information. This scale estimation is a property which is important in many applications which build on this tracking because it allows for estimation of an object's distance to the camera. To transform the detected objects from the image coordinate system to the 3D

camera coordinate system the relation of the inverse scale to the object's distance is estimated. A linear regression has to be performed once with the Least Squares Method on few known detected objects. As the scale is computed by all the features' scale of an object outliers are compensated. On e.g. the walking sequence s1 of the HumanEva [24] dataset the correlation coefficient is 0.95. Figure 2 (a) shows a 2D tracking example and Figure 2 (b) visualizes the estimated 3D information from a bird's-eye view in a rendered 3D scene.

3 Situation Recognition Subsystem

Nagel and his group developed the modelling approach of using Situation Graph Trees (SGT) to allow automatic high-level inference on the conceptual layer of situations, see [21]. An SGT is a graphical model allowing automatic high-level inference on the conceptual layer of situations. As situations are described generically, each object can instantiate different situations on its own.

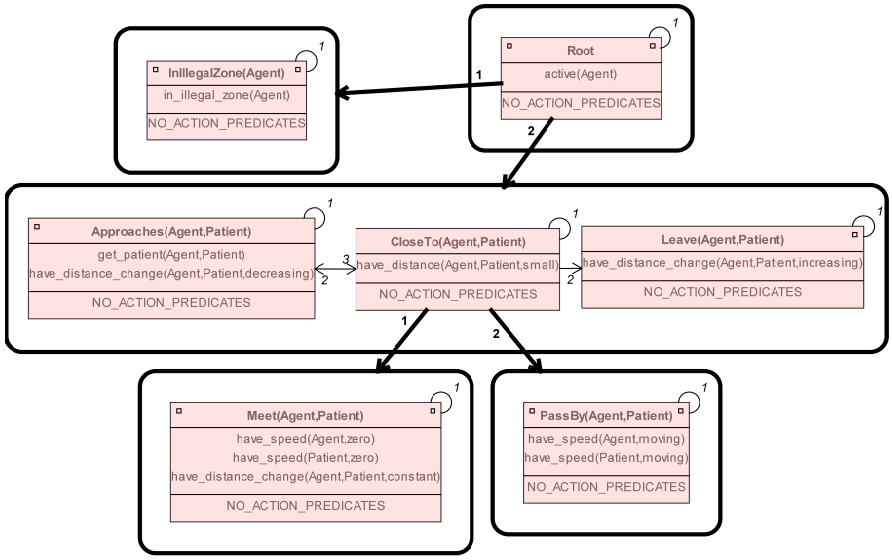


Fig. 3. An example SGT which represents the knowledge of three situations. First, the `in_illegal_zone` situation, second, `meet`, and finally, `pass_by`. A situation scheme can be conceptually (thick directed edges are specialization edges) and temporally specialized (thin directed edges are prediction edges). The priority which edge to traverse first is indicated with numbers. Small boxes in the situation scheme on the left/right indicate whether the situation scheme is a start- and/or end-scheme.

An SGT is a tree representing the knowledge about the expected behaviors of objects and consists of situation schemes, see Figure 3. Each situation scheme has a unique name, a state scheme, and an action scheme. The state scheme serves as precondition for instantiation of this situation by some *agent* object in question and consists of state predicates in Fuzzy Metric Temporal Logic (FMTL). FMTL

is an extension of first order predicate logic comprising notions of fuzzyness, time, and metrics on time, see [10]. Given a situation has been instantiated for some *agent* object the expected actions of that object are defined in the action scheme of the situation scheme. Situation schemes can have the ability to be a start and/or a final node in the SGT. Knowledge about the possible temporal development of situations can be modelled in SGTs by so-called prediction edges. They connect situations with possible succesor situations. Every situation can be specialized in a conceptual and temporal manner with an SGT again. In both cases the specialized situations are linked with specialization edges from the more general situation in a hierarchical structure. To specialize in a conceptual manner additional state predicates are added to the lower situation schemes. To specialize in a temporal manner the specialized situation is divided into several different situation schemes.

Knowledge represented in an SGT about an object is referred to as a *behavior scheme*. With such a behavior scheme it becomes possible to associate quantitative results – such as video analysis results – with conceptual knowledge.

The results of the tracking are quantitative information which have to be mapped to vague concepts with the fuzzy logic FMTL. An example of a logic predicate facilitating such a mapping is depicted in Figure 6.

Algorithm 1: Fuzzy Situation Graph Tree Traversal

```
   Input: SGT, object
1  if object occurs for the first time then
2      G ⟵ SGT root graph;
3      forall the s|s ∈ G ∧ s is start situation do
4          if s can be instantiated then
5              forall the spec|spec ∈ s ∧ spec is specialization do
6                  s := spec, G := graph containing spec;
7                  start recursion goto line 3;
8              evaluate action predicates of s;
9  else
10     forall the predSit|predSit is prediction situation of s (the last situation of
       the already known object) do
11         if instantiate predSit successful then
12             s := predSit, G := graph containing predSit;
13             start recursion goto line 3;
14         else
15             if predSit is end situation ∧ predSit ∈ G then instantiation
               successful else instantiation failed
```

The situation analysis is performed with a situation graph traversal algorithm. By now (see [3] and [11]), the situation graph traversal had found at most one instantiation of a situation for each object and each given point in time. This is not sufficient when several situations are an adequate description of an object's present situation and – moreover – when several hypotheses about an object's situation have to be considered due to uncertainty stemming from the

vision subsystem. We extended the graph traversal to a fuzzy graph traversal, see Algorithm 1. The traversal algorithm begins with the instanciation of a new object in the first start situation scheme found in the root graph(lines 3f). If the state scheme of that situation scheme can be instantiated the algorithm looks for specialization edges from this situation scheme (line 5). If they exist and lead to a start situation scheme, they are recursively instantiated (lines 6f). If from that situation scheme there are no specialization edges or the specialization cannot be instantiated the algorithm tries to proceed one time step ahead while traversing prediction edges to instantiate situation schemes (lines 10f) until an end situation is reached (lines 15f). Then the traversal of that specialization is finished and the traversal continues in the more general situation until the end node in the rood node is reached. If there are any alternatives in the specialization or temporal development of the traversal the traversal algorithm follows *all* alternatives. This means that the traversal is concurrently considering different situation schemes with different instantiations of a situation at the same time. This allows an exhaustive complete recognition of situations and their different instantiations.

4 Evaluation, Experiments, and Results

The evaluation of the situation recognition is done in three different scenarios: two with a moving infrared camera and one with a fixed surveillance camera. In the first two scenarios a driver assistance component is implemented to get automatic notifications if objects are getting too close to the vehicle. Thus, three

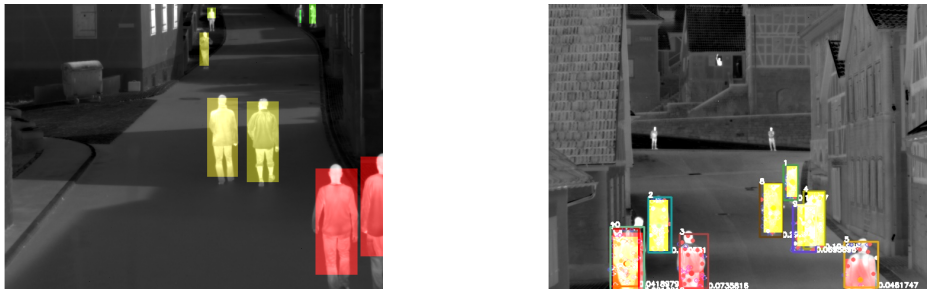


Fig. 4. Evaluation of the presented system in different scenarios. The infrared camera is mounted on a moving vehicle. The system detects and tracks persons and raises alert if persons get too close to the car.

different zones are defined by an expert. The green zone, where the objects are far away and out of interest. The orange zone, where the objects are getting closer and may become too close and the red zone, where the objects are too close to the vehicle. In Figure 4 scenes are shown where the recognized situation is visualized with colors. Based on fuzzy situations, the color is mixed due to the respective uncertainty of the recognized situations. In Figure 5 a snippet of the output of the fuzzy SGT traversal can be seen.

```
1 | 140 : 148 ! in_green_zone(obj_1).
0.766267 | 149 ! in_green_zone(obj_1).
0.233733 | 149 ! in_orange_zone(obj_1).
0.783918 | 150 ! in_orange_zone(obj_1).
0.216082 | 150 ! in_green_zone(obj_1).
0.923621 | 151 ! in_orange_zone(obj_1).
0.0763789 | 151 ! in_green_zone(obj_1).
1 | 152 ! in_orange_zone(obj_1).
```

Fig. 5. Output of FMTL facts of the traversal of the SGT. In this sequence the transition from one to another situation is shown.

```
always (is_alone(Agent,Proximity) :-
  not (exists Agent2 : (
  has_status(Agent2,_,_,_,_),
  Agent <> Agent2,
  distance_is(Agent,Agent2,Distance,Off) ,
  associate_proximity(Distance,Off,Proximity)))
).
```

Fig. 6. Predicate *is_ alone(Agent, Proximity)* in FMTL syntax. The predicate is true with *Proximity* if no other agent is closer to *Agent*.

In the third scenario the PETS2009 [7] dataset is used. Person specific situations (comprising only an *agent* object and its properties), interaction specific situations (dealing with two or more objects), and location specific situations (concerned with the relations of agents to certain places in the scene) are examined. A person specific situation is e.g. move to the left, stop, and slowly move to the right. Interaction specific situations are passing or meeting of persons and the formation of groups and their behavior. Location specific situations are e.g. the entering of an illegal zone. In Figure 3 a reduced SGT which can recognize the three situations `in_illegal_zone`, `meet`, and `pass_by` is shown. The fuzzy SGT traversal begins in the root node and tries to instantiate an agent. If this agent is not in an illegal zone the second specialization is traversed and so on. As the camera is fixed and the `in_illegal_zone` predicate proves a hard coded world model can be applied, too. Figure 7 depicts the output of this SGT for a single time point and Figure 8 evaluates our approach on PETS2009 in detail.

```
1 | 669 ! have_speed(obj_0, normal).
1 | 669 ! have_speed(obj_19, normal).
1 | 669 ! in_illegal_zone(obj_19).
1 | 669 ! have_speed(obj_22, normal).
1 | 669 ! have_speed(obj_23, normal).
1 | 669 ! close_to(obj_23,obj_26).
1 | 669 ! pass_by(obj_23,obj_26).
0.673 | 669 ! have_speed(obj_25, normal).
0.327 | 669 ! have_speed(obj_25,high).
1 | 669 ! have_speed(obj_26, normal).
1 | 669 ! close_to(obj_26,obj_23).
1 | 669 ! pass_by(obj_26,obj_23).
1 | 669 ! have_speed(obj_27, normal).
```

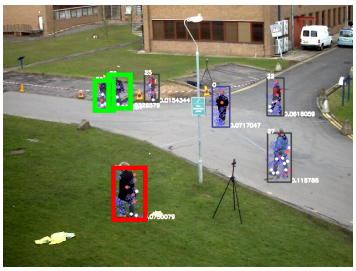


Fig. 7. Evaluation of the presented system on a scenario from PETS2009 with a fixed camera. Output of FMTL facts of the traversal of the SGT (see Figure 3) at time step 669. Situation which are recognized are `in_illegal_zone(obj_19)` (red) and `pass_by(obj_23,obj_26)` (green), see Figure 8.

5 Conclusion and Future Work

We presented a monocular vision-based fuzzy high-level situation awareness system consisting of two components. First, an object detection and tracking component providing 3D spatial and temporal information about objects in scenes

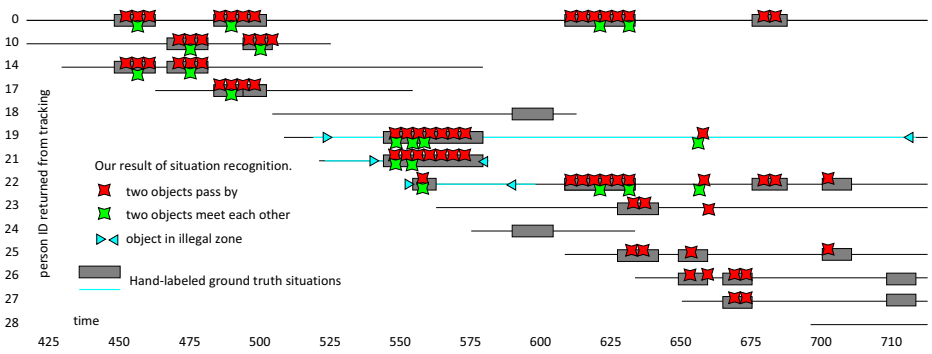


Fig. 8. Quantitative evaluation on PETS2009 data with our approach.

is presented. Second, the situation recognition is performed with knowledge encoded in SGTs. A fuzzy SGT-traversal has been introduced which allows for concurrent consideration of multiple situation hypotheses for an object at each point in time. The advantages of our methods are the possibility of using a moving camera, the availability of 3D information from the tracking, dealing with multiple objects, exchange of knowledge from different domains, and the exhaustive fuzzy SGT traversal.

The system was tested with real video data of persons and their actions and interactions. In order to show the domain independence we used recorded data from moving vehicles and static surveillance cameras. The results show that the system is usable with different spectral imagery (TV or thermal) and is simple to modify and easy to extend.

References

1. Aggarwal, J.K., Park, S.: Human motion: Modeling and recognition of actions and interactions. Proc. 3D Data Processing, Visualization and Transmission pp. 640–647 (2004)
2. Allen, J., Ferguson, G.: Actions and events in interval temporal logic. Journal of logic and computation 4(5), 531 (1994)
3. Arens, M., Gerber, R., Nagel, H.H.: Conceptual representations between video signals and natural language descriptions. IVC 26(1), 53–66 (2008)
4. Breitenstein, M., Reichlin, F., Leibe, B., Koller, E., van Gool, L.: Online multi-person tracking-by-detection from a single, uncalibrated camera. PAMI (2010)
5. Buxton, H.: Learning and understanding dynamic scene activity: a review. IVC 21(1), 125 – 136 (2003)
6. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. CVPR. vol. 1, pp. 886–893 (2005)
7. Ellis, A., Shahrokni, A., Ferryman, J.: Pets2009 and winter-pets 2009 results: A combined evaluation. In: Proc. International Workshop on Performance Evaluation of Tracking and Surveillance. pp. 1–8 (2009)
8. Enzweiler, M., Gavrila, D.M.: Monocular pedestrian detection: Survey and experiments. PAMI 31(12), 2179–2195 (2009)

9. Gammeter, S., Ess, A., Jäggli, T., Schindler, K., Leibe, B., Van Gool, L.: Articulated multi-body tracking under egomotion. In: Proc. ECCV. pp. 816–830 (2008)
10. Gerber, R., Nagel, H.H.: Representation of occurrences for road vehicle traffic. Artificial Intelligence 172(4-5), 351 – 391 (2008)
11. Gonzàlez, J., Rowe, D., Varona, J., Roca, F.X.: Understanding dynamic scenes based on human sequence evaluation. IVC, Special Section: Computer Vision Methods for Ambient Intelligence 27(10), 1433 – 1444 (2009)
12. Guerra-Filho, G., Aloimonos, Y.: A language for human action. Computer 40(5), 42–51 (2007)
13. Haritaoglu, I., Harwood, D., Davis, L.: W4s: A real-time system for detecting and tracking people in 2.5 d. In: Proc. ECCV. pp. 877–886 (1998)
14. Ivanov, Y., Bobick, A.: Recognition of visual activities and interactions by stochastic parsing. PAMI 22(8), 852–872 (2000)
15. Jüngling, K., Arens, M.: Local feature based person reidentification in infrared image sequences. In: Proc. AVSS. pp. 448–454 (2010)
16. Jüngling, K., Arens, M.: Pedestrian tracking in infrared from moving vehicles. In: Intelligent Vehicles Symposium. pp. 470–477 (2010)
17. Lavee, G., Rivlin, E., Rudzsky, M.: Understanding video events: A survey of methods for automatic interpretation of semantic occurrences in video. IEEE Trans Syst Man Cybern C Appl Rev 39(5), 489 –504 (2009)
18. Leibe, B., Leonardis, A., Schiele, B.: Robust object detection with interleaved categorization and segmentation. IJCV 77(1-3), 259–289 (2008)
19. Leibe, B., Schindler, K., Gool, L.V.: Coupled detection and trajectory estimation for multi-object tracking. In: Proc. ICCV. pp. 1–8 (2007)
20. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV 60(2), 91–110 (2004)
21. Nagel, H.H.: Steps toward a cognitive vision system. AI Mag. 25(2), 31–50 (2004)
22. Park, S., Aggarwal, J.: A hierarchical bayesian network for event recognition of human actions and interactions. Multimedia Systems 10(2), 164–179 (2004)
23. Ryoo, M., Aggarwal, J.: Semantic representation and recognition of continued and recursive human activities. IJCV 82, 1–24 (2009)
24. Sigal, L., Balan, A.O., Black, M.J.: Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. IJCV 87, 4–27 (2010)
25. Stauffer, C., Grimson, W.: Adaptive background mixture models for real-time tracking. In: Proc. CVPR. pp. 246–252 (1999)
26. Tran, S., Davis, L.: Event modeling and recognition using markov logic networks. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) Computer Vision - ECCV 2008, Lecture Notes in Computer Science, vol. 5303, pp. 610–623. Springer Berlin / Heidelberg (2008)
27. Turaga, P., Chellappa, R., Subrahmanian, V., Udrea, O.: Machine recognition of human activities: A survey. IEEE Trans. CSVT 18(11), 1473 –1488 (2008)
28. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proc. CVPR. vol. 1, pp. I–511–I–518 vol.1 (2001)
29. Vu, V.T., Bremond, F., Thonnat, M.: Automatic video interpretation: a novel algorithm for temporal scenario recognition. In: Proc. IJCAI. pp. 1295–1300 (2003)
30. Wu, B., Nevatia, R.: Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. IJCV 75(2), 247–266 (2007)