A Real-Time Scene Understanding System for Airport Apron Monitoring

No Author Given

No Institute Given

Abstract. This paper presents a distributed multi-camera visual surveillance system for automatic scene interpretation of airport aprons. The system comprises camera based tracking and classification of objects followed by sensor fusion and high level interpretation based on cognitive spatio-temporal reasoning. The performance of the system is demonstrated for a range of test scenarios.

1 Introduction

This paper describes work undertaken on the EU project AVITRACK[1]. The main aim of this project is to automate the supervision of commercial aircraft servicing operations on the ground at airports (in bounded areas known as *aprons*, shown in Figure 1). A combination of visual surveillance and video event recognition algorithms are applied in a multi-camera end-to-end system providing real-time recognition of the activities and interactions of numerous vehicles and personnel in a dynamic environment.

In visual surveillance the tracking of objects is commonly achieved using top-down(e.g. [13]) or bottom-up methods. Bottom-up tracking generally refers to a process comprising two sub-processes *motion detection* and *object tracking*; bottom-up tracking is generally computationally efficient compared to the top-down method.

Tracking algorithms have to deal with motion detection errors and complex object interactions. Apron analysis presents further challenges due to the size of the vehicles tracked (e.g. the aircraft size is $34 \times 38 \times 12$ metres), therefore prolonged occlusions occur frequently throughout congested apron operations. Many of the objects are also of near-identical appearance, consequently appearance-based matching performs poorly in such a scenario.

Video event recognition algorithms analyse tracking results spatially and temporally to automatically recognise the high-level activities occurring in the scene; for aircraft servicing analysis such activities occur simultaneously over extended time periods in apron areas. Recent work by Xiang *et al* [16] applied a hierarchical dynamic Bayesian network to recognise scene events; however, such models are incapable of recognising simultaneous complex scene activities in real-time over extended time periods. The approach adopted for AVITRACK [14] addresses these problems using cognitive vision techniques based on spatio-temporal reasoning, *a priori knowledge* of the observed scene and a set of predefined video events corresponding to aircraft service operations.

Section 2 gives an overview of the deployed system. Section 3 details the Scene Tracking module comprising per-camera motion detection, bottom-up feature-based

[1] This work is supported by the EU, grant AVITRACK (AST3-CT-3002-502818).

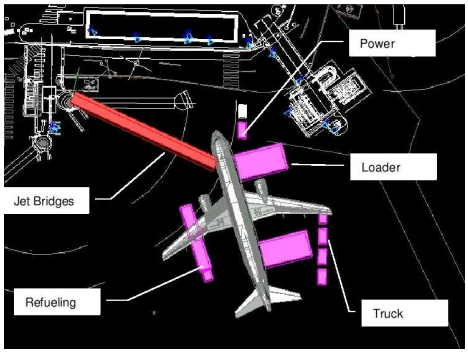

Fig. 1. The distribution of equipment around a parked aircraft in apron E40 at Toulouse Airport.

object tracking and finally fused object tracking using the combined object tracking results from the camera agents. Section 4 describes the Scene Understanding module including both the representation of video events and the video event recognition algorithm itself applied to apron monitoring. Section 5 presents the results, while Section 6 contains the discussion and lists future work.

2 System Overview

The system deployed is a decentralised multi-camera environment with overlapping fields of view (FOV); currently, eight cameras are used to monitor the scene. This system is suitable for monitoring airport aprons since there are several mounting points for cameras on the airport building and overlapping fields of view are required to ensure consistent object labelling and enhanced occlusion reasoning within the scene. The majority of the camera mounting points observe the right hand side of the fuselage since this is where baggage loading and unloading operations take place. Spatial registration of the cameras is performed using per camera coplanar calibration and the camera streams are synchronised temporally across the network by the central video server.

The architecture of the system is shown in Figure 2 comprising two main modules - *Scene Tracking* and *Scene Understanding*. In the Scene Tracking module a *Frame Tracker* module runs independently for each of the cameras, performing motion detection, frame to frame tracking and object categorisation. A central *Data Fusion* module receives the single-camera observations from the Frame Tracker modules, fuses the observations and generates 3D results to maximise the useful information content of the scene being observed. In the Scene Understanding module a *Long-Term Tracker* uses a temporal window to provide trajectory information required for event recognition and behaviour analysis. The *Event and Scenario Recognition* module uses the tracking results to perform event detection and high level scene interpretation. An offline *Scene Modelling* module is used to generate geometric and semantic scene and object models, as well as defining video event models.

The system must be capable of monitoring and recognising the activities and interaction of numerous vehicles and personnel in a dynamic environment over extended

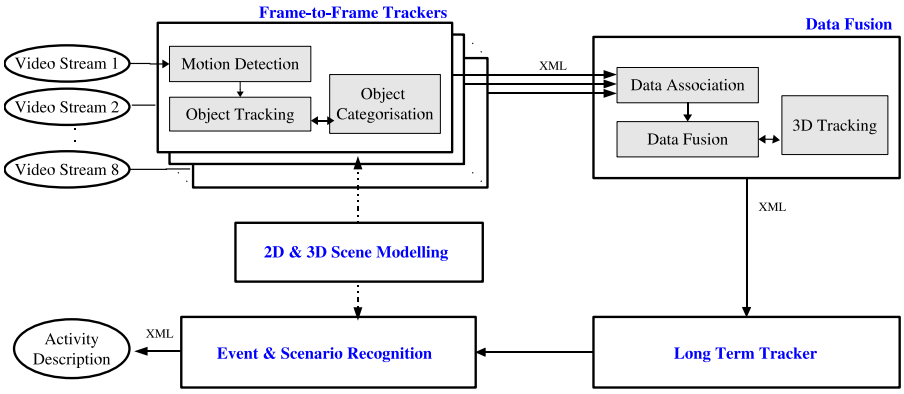

Fig. 2. The system architecture deployed for the AVITRACK project.

periods of time, operating in real-time (12.5 FPS, 720×576 resolution) on colour video streams. The relatively low quantity of the distributed modules and the physical distances between them allows the network to be operated via a standard 1Gb ethernet.

The communications framework selected for the distributed modules is via a ORO-COS::SmartSoft CORBA [11] implementation. The modules communicate using the XML standard; although inefficient for communication over a network, the XML standard allows the system to be efficiently integrated as a series of black box modules with a defined interface between them. The partners in the project are able to develop the modules independently while adhering to the XML interface standard; this standardisation allowed the modules to be successfully integrated in the end-to-end system with few problems. The added advantage of the XML is that the human operators can manually inspect the XML to explain some system failures that may occur during integration.

3 Scene Tracking

The Scene Tracking module is responsible for the per-camera detection and tracking of moving objects, transforming the image positions into 3D world co-ordinates, and fusing the multiple camera observations of each object into single world measurements.

3.1 Frame-to-Frame Tracking

For detecting connected regions of foreground pixels, 16 motion detection algorithms were implemented for AVITRACK and evaluated quantitatively on various apron sequences under different environmental conditions (sunny conditions, fog, etc.). The metrics adopted, the evaluation process and the results obtained are described in more detail in [1]. Taking into account processing efficiency as well as sensitivity, the colour mean and variance method was selected [15]. This motion detector has a background model represented by a pixel-wise Gaussian distribution $N(\mu, \sigma^2)$ over the normalised RGB colour space. In addition, a shadow/highlight detection component based on the work of Horprasert *et al* [10], is used to handle illumination variability. The algorithm

also employs a multiple background layer technique to allow the temporary inclusion into the background model of objects that become stationary for a short period of time. For real-time object tracking, the KLT algorithm [12] is used, and it considers features to be independent entities and tracks each of them individually. Therefore, it is incorporated into a higher-level tracking process that groups the sparse local features into objects, maintain associations between features and objects, and uses the individual tracking results of the features to track the objects globally, while taking into account complex object interactions. To mantain the association between features and objects from one frame to the next, the spatial information and the motion information of features are used. Spatial rule-based reasoning is applied to detect the presence of merging or splitting foreground regions, based on the idea that if a feature belongs to an object at time $t-1$, then the feature should remain spatially within the foreground region of the object at time t. The motion of the individual features are robustly fitted to translational and affine motion models to estimate the membership of features to objects. If the motion models are not distinct or unreliable then spatial-based reasoning is used; otherwise a combination of both is used.

On the apron activity tends to happen in congested areas with several vehicles stationary in the proximity of the aircraft. To allow stationary and moving objects to be differentiated, the motion detection process (Section 3.1) was extended to include a multiple background layer technique. The tracker identifies stopped objects by one of two methods: analysing an object's region for connected foreground pixels which have been labelled as 'motion' over a time window; or by checking the individual motion of local features of an object. The accuracy of the second method depends on the feature density parameter $\boldsymbol{\rho}$. Stationary objects are integrated into the motion detector's background model as different background layers. The advantage this method over pixel level analysis (e.g. Collins *et al* [7]) is that for extended time periods (e.g. 30 minutes) pixel level methods tend to result in fragmented layers that do not represent cohesive objects. More detail about the Scene Tracking module can be found in [2].

To efficiently recognise the people and vehicles on the apron, a multi-stage categorisation approach is adopted: the first stage consists of a bottom-up process that categorises the main object categories (people, ground vehicles, aircraft or equipment); this is achieved using a Gaussian mixture model classifier trained on efficient descriptors such as 3D width and height, dispersedness and aspect ratio. This is inspired by the work of Collins *et al* [6] where it was shown to work well for distinct object classes. The classification stage is applied to the vehicle category to recognise the individual vehicle sub-types, which cannot be determined from simple descriptors. Hence, a proven top-down method [9, 13] is applied to fit textured 3D models to the detected objects in the scene. These models are fitted to the image data by back projection and evaluated using normalised cross-correlation to determine the best model pose.

3.2 Data Fusion

The Data Fusion module is based on a nearest neighbour Kalman filter approach [4] with a constant velocity model. The measurement uncertainty is estimated by propagating a nominal image uncertainty using the method presented in [5]. The measurement uncertainty field is shown in Figure 3 for camera 6; this estimate of uncertainty al-

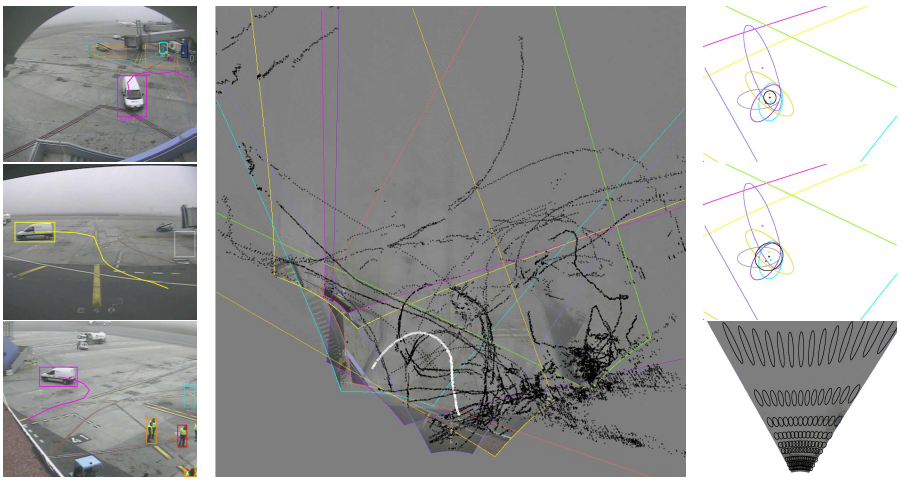

Fig. 3. (Left) Tracking results for 3 cameras for frame 9126 of sequence 21. (Middle) shows data fusion results on the ground-plane for the sequence (9600 frames) with the vehicle track shown in white. (Top-right) the fused observation (in black) for the vehicle (frame 9126) using the covariance accumulation method, (Middle-right) shows the result for covariance intersection. (Bottom-right) shows the sensory uncertainty field measured for camera 6.

lows formal methods to be used to associate observations originating from the same measurement, as well as providing mechanisms for fusing observations into a single estimate.

In the association step a validation gate [4] is applied to limit the potential matches between tracks and observations. Matched observations are fused to find the estimate of the location and uncertainty of the object, based on covariance intersection. Covariance intersection estimates the fused uncertainty for a set of matched observations as a weighted inverse summation; the weighting is chosen such that it is in favour of the sensors that have more certain measurements. The fused observations are demonstrated in Figure 3; the (unweighted) covariance accumulation method [5] results in a more localised estimate of the fused measurement than the covariance intersection approach. More detail about the Data Fusion module can be found in [2].

4 Scene Understanding

The Scene Understanding module is responsible for the recognition of video events in the scene observed through video sequences. This module performs a high-level interpretation of the scene by detecting video events occurring in it. The method to detect video events uses cognitive vision techniques based on spatio-temporal reasoning, *a priori* knowledge of the observed environment and a set of predefined event models which are written using the description language described in [8]. A Video Event Recognition module takes the tracked mobile objects from the previously described modules as input, and outputs video events that have been recognised. The *a priori* knowledge

exploited includes the camera information, the vehicle models, the expected moving objects and the empty scene model containing the contextual objects.

4.1 Video Event Representation

The video event representation corresponds to the specification of all the knowledge used by the system to detect video events occurring in the scene. To allow experts in the aircraft activity monitoring to easily define and modify the video event models, the description of the knowledge is declarative and intuitive (in natural terms). The video event representation is based on the video event description language described in [8]. Thus, the video event recognition uses the knowledge represented by experts through event models. The proposed model of a video event E is composed of five parts:

- a set of Physical Object variables corresponding to the physical objects involved in E: any contextual object including static object (equipment, zone of interest) and mobile object (person, vehicle, aircraft, etc.) The vehicle mobile objects can be of different subtypes to represent different vehicles (GPU, Loader, Tanker, etc.)
- a set of temporal variables corresponding to the components (sub-events) of E.
- a set of forbidden variables corresponding to the components that are not allowed to occur during the detection of E.
- a set of constraints (symbolic, logical, spatial and temporal constraints including Allen's interval algebra operators [3]) involving these variables.
- a set of decisions corresponding to the tasks predefined by experts that need to be executed when E is detected (e.g. activating an alarm or displaying a message).

There are four types of video events: primitive state, composite state, primitive event and composite event. A state describes a situation characterising one or several physical objects defined at time t or a stable situation defined over a time interval. A primitive state (e.g. a person is inside a zone) corresponds to a vision property directly computed by the vision module. A composite state, as shown in Figure 4, corresponds to a combination of primitive states. An event is an activity containing at least a change of state values between two consecutive times (e.g. a vehicle leaves a zone of interest - it is inside the zone and then it is outside). A primitive event, as shown in Figure 4, is a change of primitive state values and a composite event is a combination of states and/or events.

```
CompositeState(Vehicle_Stopped_Inside_Zone,
PhysicalObjects((v1 : Vehicle), (z1 : Zone))
Components( (c1 : PrimitiveState Inside_Zone(v1, z1))
            (c2 : PrimitiveState Vehicle_Stopped(v1)))
Constraints(  (c2 during c1)))
```

```
PrimitiveEvent(Enters_Zone,
  PhysicalObjects((m1 : MobileObject), (z1 : Zone))
  Components( (c1 : PrimitiveState Outside_Zone(m1, z1))
              (c2 : PrimitiveState Inside_Zone(m1, z1))
  Constraints(  (c1 meet c2)))
```

Fig. 4. (Left) The model of the composite state "Vehicle_Stopped_Inside_Zone": a vehicle is detected as stopped inside a zone of interest. (Right) The model of the primitive event "Enters_Zone": a vehicle enters a zone of interest.


```
CompositeEvent(Unloading_Operation,
  PhysicalObjects( (p1 : Person), (v1 : Vehicle), (v2 : Vehicle), (v3 : Vehicle),
                   (z1 : Zone), (z2 : Zone),(z3 : Zone), (z4 : Zone))
  Components(  (c1 : CompositeEvent Loader_Arrival(v1, z1, z2))
               (c2 : CompositeEvent Transporter_Arrival(v2, z1, z3))
               (c3 : CompositeState Worker_Manipulating_Container(p1, v3, v2, z3, z4)))
  Constraints( (v1->SubType = LOADER)
               (v2->SubType = TRANSPORTER)
               (z1->Name = ERA)
               (z2->Name = RF_DoorC_Access)
               (z3->Name = LOADER_BackZone)
               (z4->Name = TRANSPORTER_BackZone)
               (c1 before_meet c2)
               (c2 before_meet c3)))
```

Fig. 5. (Left) Two dynamic zones (in blue) linked with the Loader and the Transporter vehicles involved in the detected event "Worker_Manipulating_Container" (event 26). (Right) The Unloading operation involves 8 physical objects and 3 composite components with 2 constraints on the vehicle subtypes, 4 constraints on the zones of interest and 2 temporal constraints.

4.2 Video Event Recognition

The video event recognition algorithm recognises which events are occurring in a stream of mobile objects tracked by the vision module. The recognition of composite states and events usually requires a search in a large space composed of all the possible combinations of components and objects. To avoid this combinatorial explosion, all composite states and events are simplified into states and events composed of at most 2 components through a stage of compilation in a preprocessing phase. Then the recognition of composite states and events is performed in a similar way to the recognition of primitive events, as described in the method of Vu *et al* [14].

In the Video Event Recognition module, *a priori* knowledge corresponds to apron zones of interest (access zones, stopping zones), aircraft and vehicle (e.g. GPU, Loader, Tanker and Transporter) models. In apron monitoring, some problems may occur while trying to build an accurate context of the scene. For example, access zones to aircraft can be at different positions according to the aircraft type. To solve these problems, dynamic properties have been added to the *a priori* knowledge, by defining dynamic zones in the local coordinate system of vehicles. Figure 5 illustrates the use of dynamic context. This notion of dynamic context allows more complex scenarios to be defined in which mobile objects can directly interact with each other.

4.3 Predefined Video Events

Currently a set of 21 basic video events have been defined, including 10 primitive states, 5 composite states and 6 primitive events. These basic video events are used in the definition of video events representing the handling operations. The primitive states correspond to spatio-temporal properties related to persons and vehicles involved in the scene. Some examples include: a person is located inside a zone of interest, a person is close to a vehicle, a vehicle is located inside a zone of interest, a vehicle is close to another vehicle, a vehicle has stopped, etc. Using these primitive states, different

composite states have been modelled, such as: a person stays inside a zone of interest, a vehicle has arrived in a zone of interest, and a vehicle has stopped in a zone of interest (shown in Figure 4). The composite states have in turn been used to model different primitive events, for example: a person enters a zone of interest, a person moves between zones of interest, a vehicle enters a zone of interest (shown in Figure 4), a vehicle moves between zones of interest, etc. These states and events are then used in the definition of the composite events (modelling behaviours) representing the apron operations.

Current work is on video events involving (1) the GPU (Ground Power Unit) vehicle which operates in the aircraft arrival preparation operation, (2) the Tanker vehicle which operates in the refuelling operation and (3) the Loader and Transporter vehicles which are involved in the baggages loading/unloading operations. To recognise these operations 28 composite video events were defined, including 8 video events for the aircraft arrival preparation operation, 8 video events for the refuelling operation, and 12 video events for the unloading operation.

The aircraft arrival preparation operation (event 8) involves the GPU, its driver and 4 zones of interest. The system recognises that the GPU vehicle arrives in the ERA Zone (event 1), obeys the speed limit (event 2); then it enters (event 3) and stops (event 4) in the "GPU Access Area", the driver gets out of the vehicle (event 5) and deposits the chocks and stud at the location where the plane will stop (events 6 and 7). This operation and another modelled one, the refuelling operation, are considered to be basic operations because they only consist of one person and one vehicle.

The baggage unloading operation (Figure 5) is more complex. This operation involves both a Loader and a Transporter vehicle, the conductor of the Loader, and a person working in the area. This operation is composed of the following steps: first, the Loader vehicle arrives in the ERA zone (event 17), enters its restricted area (event 18) and then stops in this zone (event 19); a dynamic zone is automatically added, at the rear of the Loader's stop position ("Loader_Arrival", event 20), where the Transporter will enter and stop. When the Transporter enters (event 21) and stops (event 22) in this zone ("Transporter_Arrival", event 23), another dynamic zone is automatically added to the context. The back of the Loader is then elevated (event 24) and the baggage containers are unloaded from the aircraft by the Loader conductor (event 25) one by one. The conductor unloads these containers into the dynamic zone of the Transporter where a worker arrives (event 26) and directs the containers (event 27) on to the Transporter.

5 Results

The Scene Tracking evaluation assesses the performance of the three core components (motion detection, object tracking and data fusion) on representative test data.

The performance evaluation of the different motion detector algorithms for AVITRACK is described in more detail in [1]. It is noted that some objects are partially detected due to the achromaticity of the scene and the presence of fog causes a relatively high number of foreground pixels to be misclassified as highlighted background pixels resulting in a decrease in accuracy. Strong shadows also cause problems, often detected as part of the mobile objects. The performance evaluation of the tracking algorithm is described in more detail in [2]. In is noted that some objects can produce a ghost which

remains behind the previous object position. An object is integrated into the background when becomes stationary for an extended time period. In these cases, ghosts are created when stationary objects start to move again. Partial detection of objects can result in fragmentation in tracked objects with similar colour as the background.

The Data Fusion module performs adequately given correctly detected objects in the Frame Tracker (a representative result is shown in Figure 3). The Data Fusion module incorporates uncertainty information in the location estimate of the observation and it is often an inaccurate location estimate that results in the failure of the data association step; a significant proportion of the localisation problems that occur in the Data Fusion module can be traced back to motion detection errors i.e. shadow, reflections etc.

The Scene Understanding evaluation have been performed on sequences for which the tracking module gives good results. Video event recognition has been tested on sequences involving the GPU (aircraft arrival preparation operation), the Tanker (refuelling operation) and the Loader/Transporter vehicles (baggage unloading operation).

Video events 1 to 4 involving a GPU have been tested on a dataset of 4 scenes corresponding to 2×4 video sequences (containing from 1899 to 3774 frames and including one night sequence). These events are detected with a perfect True Positive rate. The video events 4 to 8 involving also a GPU have been tested on 2 scenes corresponding to 2 video sequences because only one camera is available to observe these events. The video events involving the Tanker have been tested on one scene (more than 15000 frames corresponding to about 30 minutes) showing the "Tanker Arrival" (event 13) and the driver of the Tanker branching the refuelling pipe to the aircraft (events 14, 15, 16). The "Unloading Baggage operation" involving the Loader (events 17 to 20, event 24 and event 25) and the Transporter (events 21 to 23) have been tested on one scene where the cameras point of view allows to fully observe the vehicle movements and interactions between vehicles and people.

The results of the qualitative evaluation are shown in Table 2. The goal is to give an idea of the performance of the Scene Understanding and to anticipate potential problems in event detection for apron monitoring. All video events are recognised correctly (49 TPs) without false alarms (0 FPs) and misdetection (0 FNs). These results are very encouraging but one has to keep in mind that situations where the vision module misdetects or overdetects mobile objects were not addressed.

Vehicle type	**Sequence**	**TP**	**FP**	**FN**
GPU				
Events 1 to 4	4 scenes * 2 cam.	32	0	0
Events 4 to 8	2 scenes * 1 cam.	8	0	0
Tanker				
Events 9 to 13	2 scenes * 1 cam.	10	0	0
Events 14 to 16	1 scene * 1 cam.	3	0	0
Loader-Transporter				
Events 17 to 28	1 scenes * 1 cam.	12	0	0

Table 1. Performance results of the Scene Understanding module for apron monitoring. TP = "Event exists in the real world and is well recognised", FN = "Event exists in the real world but is not recognised", FP = "Event does not exist in the real world but is recognised".

6 Discussion and Future Work

The results are encouraging for the presented system. The performance of multi-view object tracking provides adequate results; however, tracking is sensitive to significant dynamic and static occlusions within the scene. Future work will address shadow/ghost supression and explicit occlusion analysis.

The Scene Understanding results show that the proposed approach is adapted to apron monitoring and can be applied to complex activity recognition. The recognition of complex operations in parallel (e.g. "baggage unloading") involving people and vehicles gives encouraging results. Future work will incorporate uncertainty to enable recognition of events even when the Scene Tracking module gives unreliable output.

References

1. Reference removed for anonymity.
2. Reference removed for anonymity.
3. J. F. Allen. Maintaining knowledge about temporal intervals. In *Communications of the ACM*, volume 26 num 11, pages 823–843, Nov 1983.
4. Y. Bar-Shalom and X.R. Li. *Multitarget-Multisensor Tracking: Principles and Techniques*. YBS Publishing, 1995.
5. J. Black and T.J. Ellis. Multi Camera Image Measurement and Correspondence. In *Measurement - Journal of the International Measurement Confederation*, volume 35 num 1, pages 61–71, 2002.
6. Collins, Lipton, Kanade, Fujiyoshi, Duggins, Tsin, Toliver, Enomoto, and Hasegawa. A System for Videosurveillance and Monitoring: VSAM Final Report. Technical Report CMU-RI-TR-00-12, Robotics Institute, Carnegie Mellon University, May 2000.
7. R. Collins, A. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, O. Hasegawa, P. Burt, and L. Wixson. A system for video surveillance and monitoring. In *Tech. Report CMU-RI-TR-00-12*, May 2002.
8. M. Thonnat F. Brémond, N. Maillot and V. Vu. Ontologies for video events. In *Research report number 51895*, Nov 2003.
9. J. M. Ferryman, A. D. Worrall, and S. J. Maybank. Learning enhanced 3d models for vehicle tracking. In *Proc. of the British Machine Vision Conference*, 1998.
10. T. Horprasert, D. Harwood, and L.S. Davis. A statistical approach for real-time robust background subtraction and shadow detection. In *IEEE ICCV'99 FRAME-RATE Workshop*, 1999.
11. C. Schlegel. A component approach for robotics software: Communication patterns in the orocos context. In *18. Fachtagung Autonome Mobile Systeme (AMS), Informatik aktuell*, pages 253–263. Springer, Karlsruhe, Dec 2003.
12. J. Shi and C. Tomasi. Good features to track. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 593–600, 1994.
13. G. D. Sullivan. Visual interpretation of known objects in constrained scenes. In *Phil. Trans. R. Soc. Lon.*, volume B, 337, pages 361–370, 1992.
14. V. Vu, F. Brémond, and M. Thonnat. Automatic video interpretation: A novel algorithm for temporal event recognition. In *IJCAI'03, Acapulco, Mexico*, Aug 2003.
15. C. R. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: Real-time tracking of the human body. In *IEEE Transactions on PAMI*, volume 19 num 7, pages 780–785, 1997.
16. T. Xiang and S. Gong. On the structure of dynamic bayesian networks for complex scene modelling. In *Proc. Joint IEEE Int. Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS)*, pages 17–22, Oct 2003.