A Real-Time Scene Understanding System for Airport Apron Monitoring

David Thirde, Mark Borg and James Ferryman
Computational Vision Group, The University of Reading, UK

Florent Fusier, Valery Valentin,
Francois Bremond and Monique Thonnat
ORION Team, INRIA Sophia-Antipolis, France

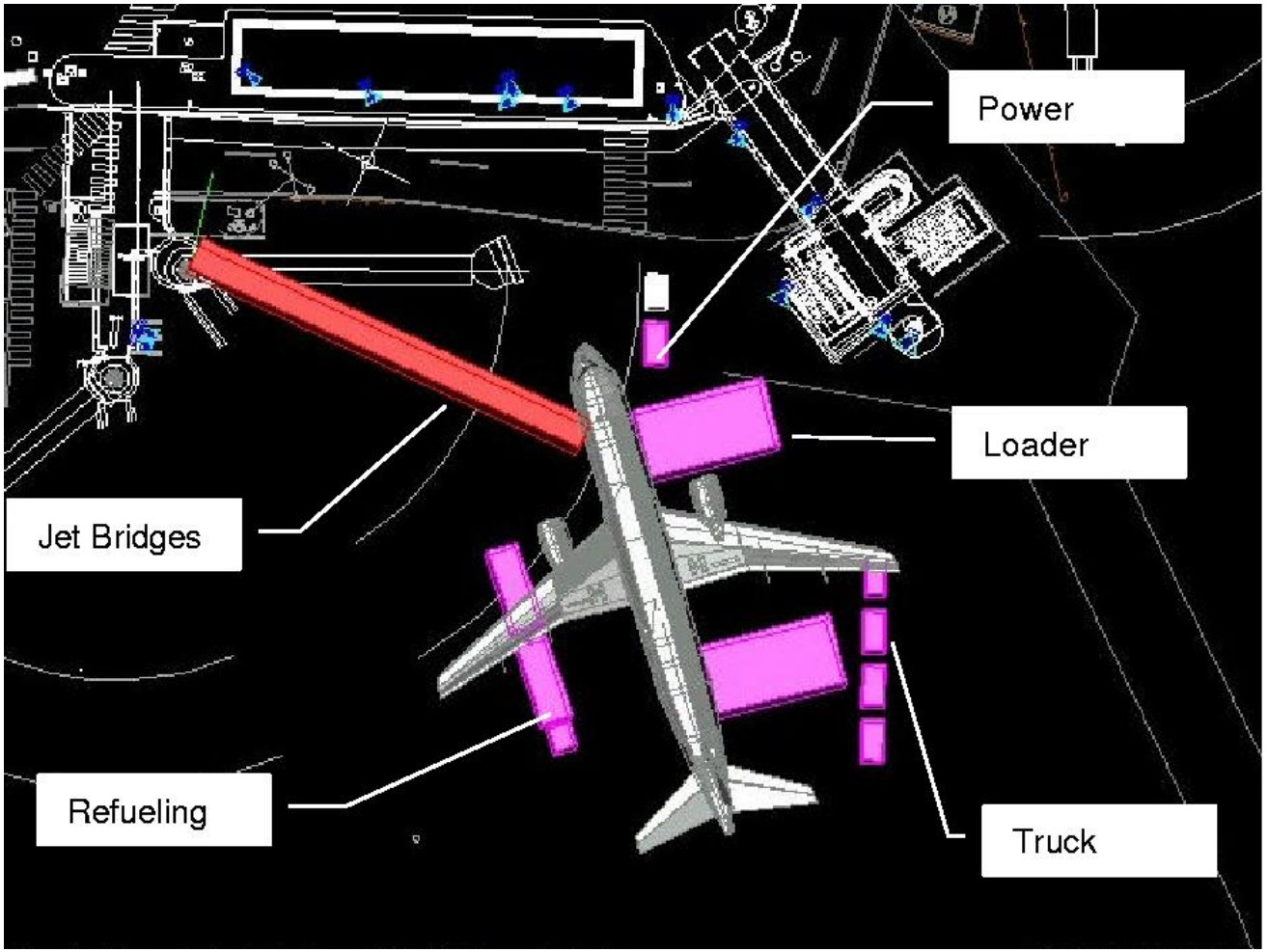
AVITRACK

Aircrafts surroundings, categorised Vehicles & Individuals Tracking for apRon's Activity model interpretation & CheckK

- **Objective:** to automate recognition of activities around parked aircraft on apron areas to improve competitiveness, safety and security

- **Scope:** develop a distributed vision system performing multi-camera visual surveillance and event recognition in real-time.

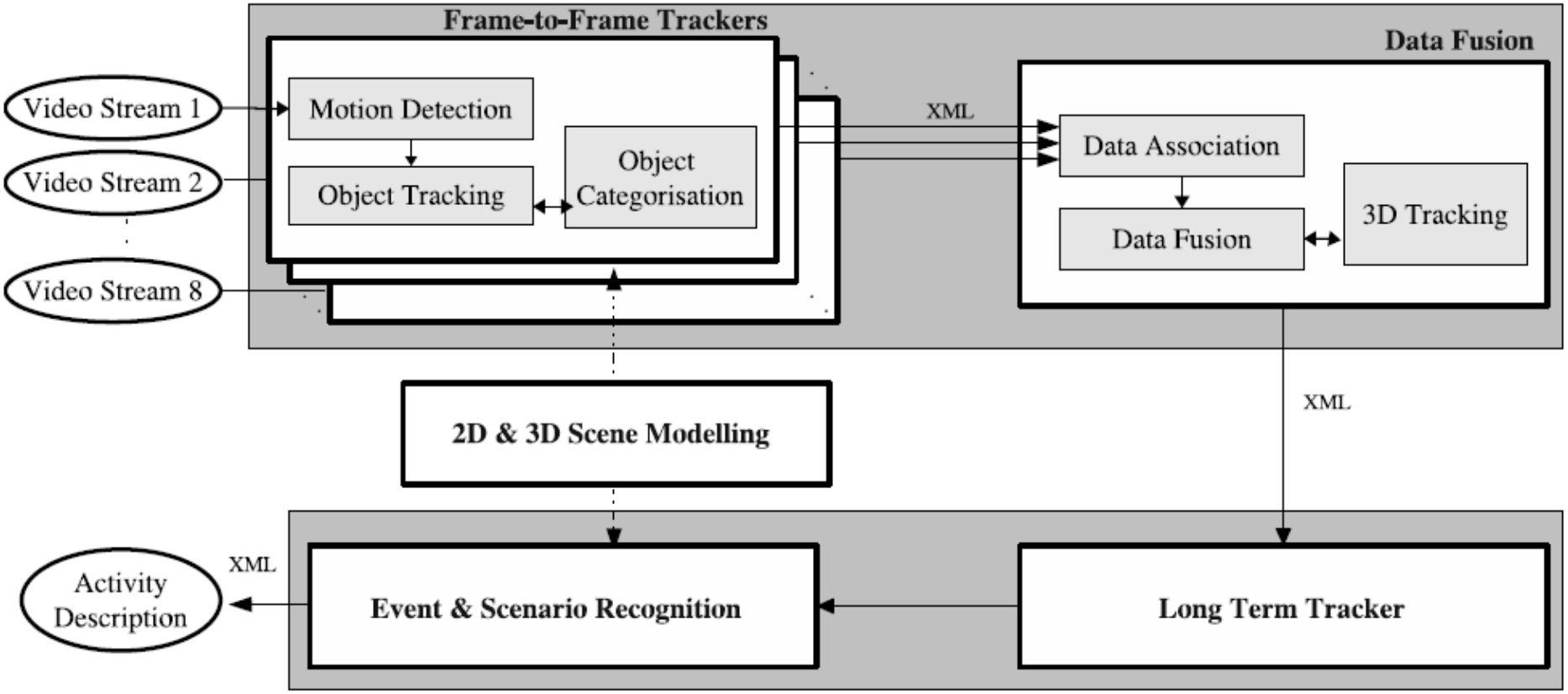
Apron E-40, Toulouse Airport, France



Application Constraints

- **The system must:**
  - Monitor and recognise the interaction of numerous vehicles and personnel
  - Operate in a dynamic environment over extended time periods
  - Operate in real-time (defined as 12.5 FPS for PAL colour images)

Architecture



Overview

- Frame Trackers (per-camera)
  - Motion Detection
  - Object Tracking
  - Object Recognition
- Data Fusion
- Long Term Tracker
- Event and Scenario Recognition

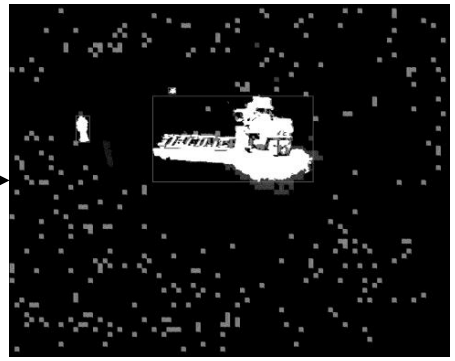
Overview

- Frame Trackers (per-camera)
  - Motion Detection
  - Object Tracking
  - Object Recognition
- Data Fusion
- Long Term Tracker
- Event and Scenario Recognition

Motion Detection

Video data feed

Motion Detection



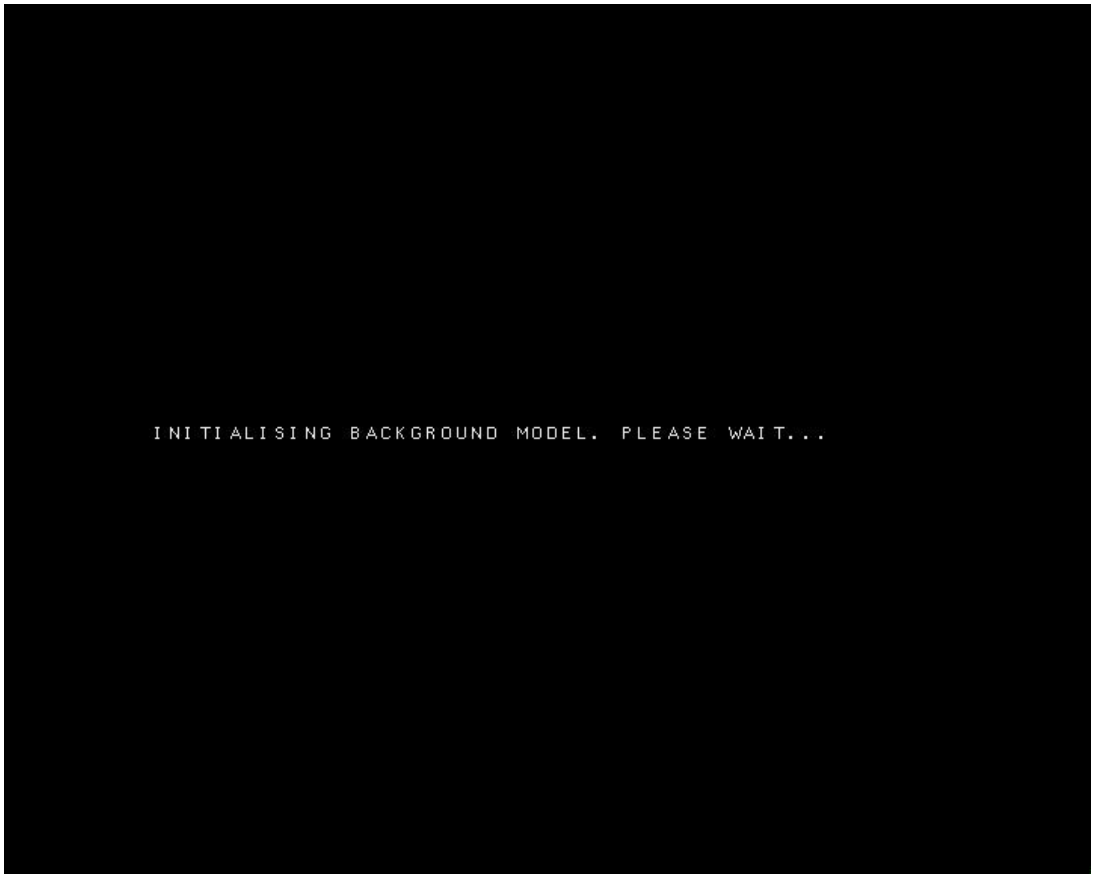
2D blobs

Motion Detection

- Background subtraction using pixel-wise Gaussian background model in normalised RGB colour space
- Object based background layering to allow moving objects to be differentiated from stationary objects.
- Shadow and highlight suppression module based on work of Horprasert *et al* (ICCV'99)

Motion Detection Result

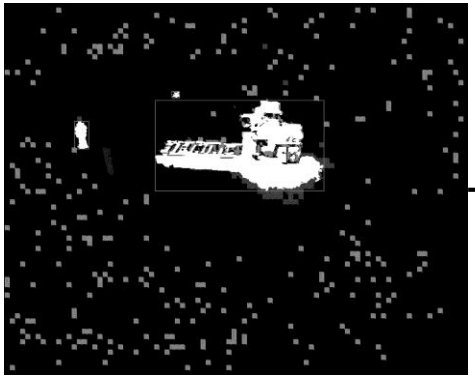
Motion Detection



Overview

- Frame Trackers (per-camera)
  - Motion Detection
  - Object Tracking
  - Object Recognition
- Data Fusion
- Long Term Tracker
- Event and Scenario Recognition

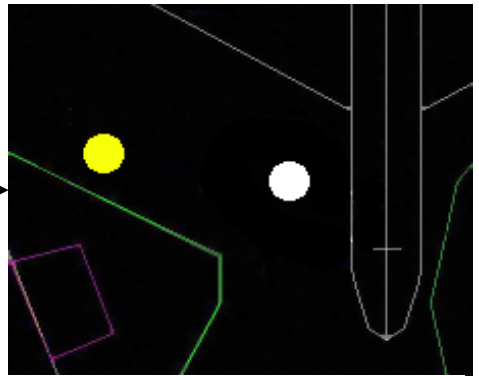
Object Tracking



2D blobs

Object Tracking

2D tracks



3D localisation

KLT Tracking

1. **KLT interest points:** generated for object regions.
   - Each point has object membership in frame t-1.
   - Each object contains >1 features.

2. **KLT algorithm:** Track points to current frame t

3. **KLT features:** Match to frame t motion regions and handle interactions.

4. **Replenish** KLT interest points and continue.

KLT Tracking II

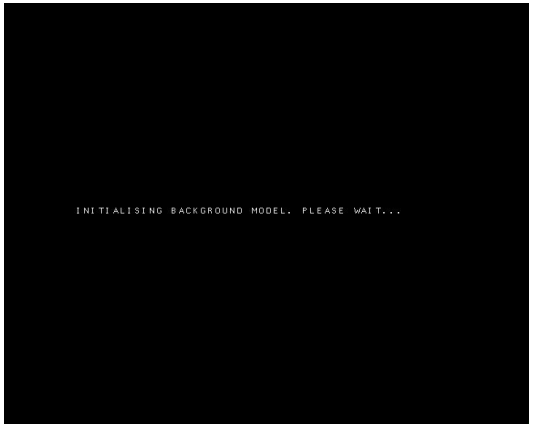
- **Match function:** determines if motion region correspondence is
  - One-to-one (tracked)
  - One-to-many (split)
  - Many-to-one (merge)
  - One-to-none (missing)
  - None-to-one (new)

- When merging, an object's state is predicted by fitting a translational motion model to the tracked points.

- To detect splitting of objects the intra-object interest points are robustly clustered using translational motion models.

Object Tracking Result

Object Tracking Result

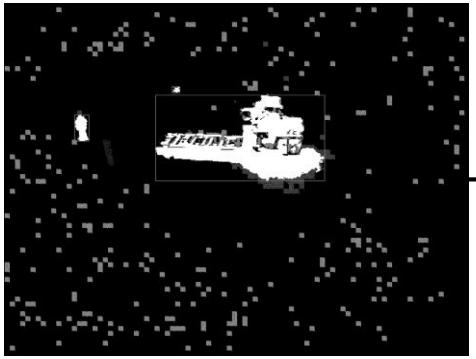


Overview

- Frame Trackers (per-camera)
  - Motion Detection
  - Object Tracking
  - Object Recognition
- Data Fusion
- Long Term Tracker
- Event and Scenario Recognition

Object Recognition

2D tracks



2D blobs

Object Recognition

Aircraft	**0.1**
Transporter	**0.7**
GPU	**0.1**
Service Vehicle	**0.1**
Person	**0.0**
Other	**0.0**

Per track category vectors

Object Recognition

- Multi-stage classifier combining efficient bottom-up and expensive top-down classifiers:
  - Stage One: classify main object types (people, vehicles, aircraft etc) using a GMM trained on efficient descriptors (3D width/height etc)
  - Stage Two: for vehicle type, classify sub-type (loader, tanker etc) using textured 3D models fitted using NCC and SIMPLEX search within the parameter space.

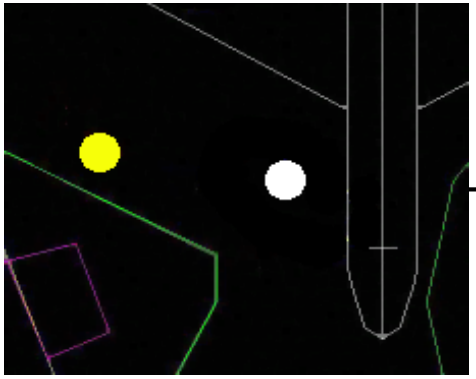
Object Recognition Result

Overview

- Frame Trackers (per-camera)
  - Motion Detection
  - Object Tracking
  - Object Recognition
- Data Fusion
- Long Term Tracker
- Event and Scenario Recognition

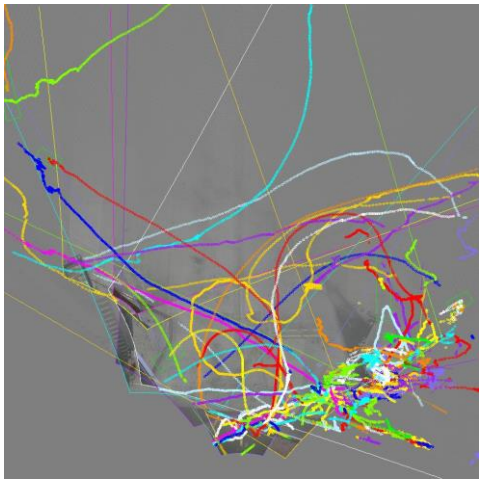
Data Fusion

Per-camera 2D tracks



Per-camera 3D localisation

Data Fusion



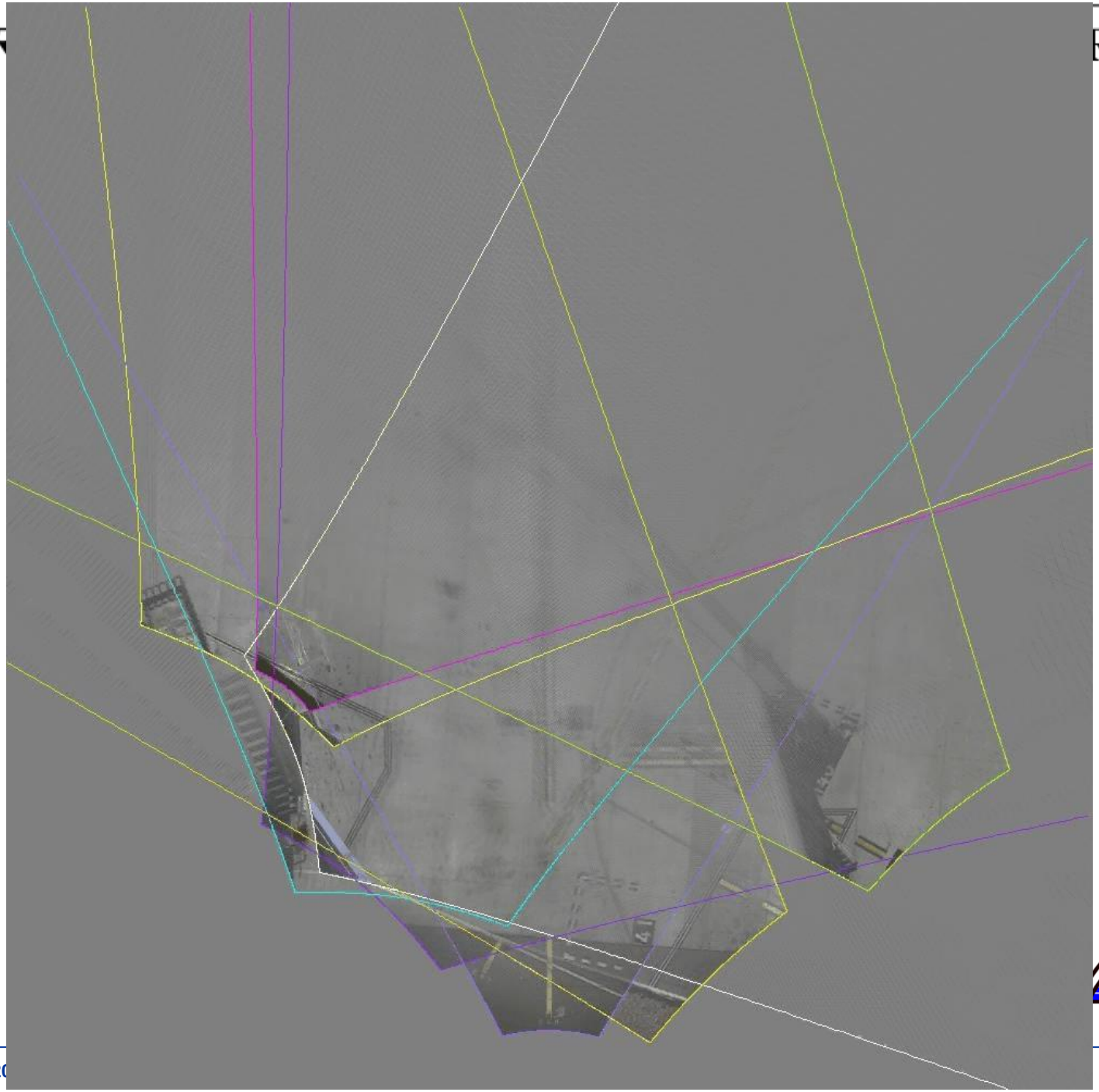
3D fused tracks

Data Fusion

- Discrete Nearest Neighbour Kalman Filter approach with constant velocity

1. Validation gate used to limit the potential matches between tracks and per-camera measurements.
2. Data association: nearest neighbour per camera to a track.
3. Fusion of associated measurements.
4. Kalman filter update of each track state with fused measurement.
5. New candidate tracks from remaining measurements.

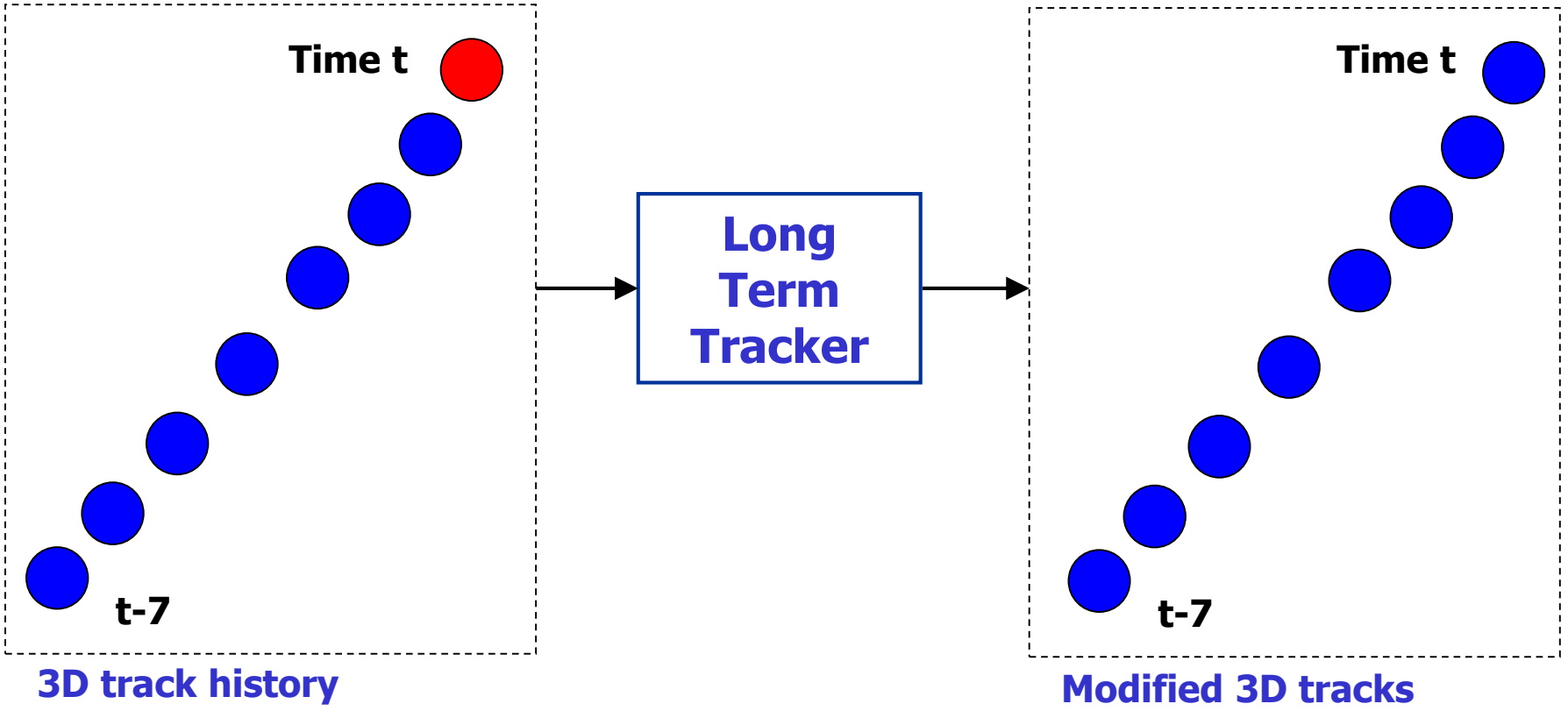
Data Fusion Result



Overview

- Frame Trackers (per-camera)
  - Motion Detection
  - Object Tracking
  - Object Recognition
- Data Fusion
- Long Term Tracker
- Event and Scenario Recognition

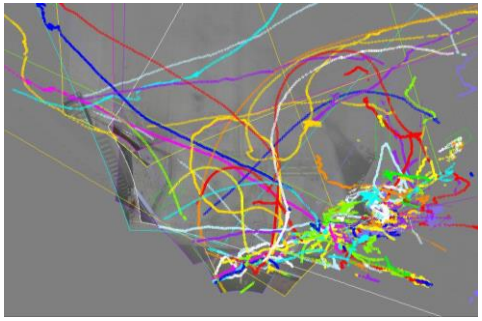
Long Term Tracker



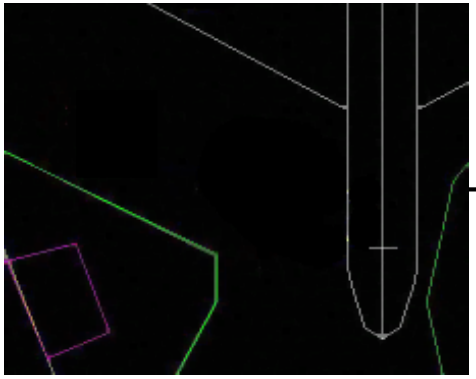
Overview

- Frame Trackers (per-camera)
  - Motion Detection
  - Object Tracking
  - Object Recognition
- Data Fusion
- Long Term Tracker
- Event and Scenario Recognition

Event and Scenario Recognition

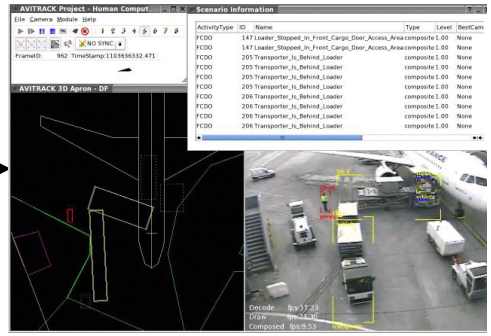


3D tracks



Contextual knowledge / Event Models

Event and Scenario Recognition



Recognised apron activities (displayed by HCI)

Event and Scenario Recognition

- **Approach:** Event Recognition based on
  - a priori knowledge of the observed environment
  - models of events predefined by application domain experts
  - spatio-temporal reasoning based on temporal constraints propagation

3D Scene Model of the Observed Environment

- **Definition:** a priori knowledge of the observed empty scene
  - **Cameras** e.g. intrinsic, extrinsic parameters
  - **3D Geometry** of moving/static objects and ground plane zones e.g. location, shape, volume
  - **Semantics**:
    - type (e.g. object, zone)
    - characteristics (e.g. appearance)
    - function (e.g. seat)

3D Scene Model of the Observed Environment

- **Purpose:**
  - keep the interpretation independent from the sensors and the sites: many sensors, one 3D referential
  - provide additional knowledge for activity recognition

Video Event Representation: Video Event Model

- We have defined four types of video events:

primitive state	**composite state**
primitive event	**composite event**

- A video event is constituted by three parts:
  - **Physical objects:** all real world scene objects
  - **Components:** list of states and events
  - **Constraints:** symbolic, logical, spatio-temporal relations between components or physical objects

Video Event Representation

- **States:** describe situations characterising one or more objects defined at a time instance or a stable situation defined over a time interval
  - **Primitive State**: a measurement computed from the tracking module output (e.g. a person is inside a zone)
  - **Composite State**: a combination of primitive states

Video Event Representation

- **Examples of states:**

A person is inside the Equipment Restricted Area (ERA) zone

The Ground Power Unit (GPU) Vehicle is stopped in GPU access area

Video Event Representation

- **Events:** activities containing at least a change of state values between two consecutive times
- **Primitive Event:** corresponds to a change of primitive state values

  - **Example:** The Ground Power Unit (GPU) Vehicle enters in the GPU access area

Video Event Representation

- **Composite Event:** corresponds to a combination of states and events i.e. a *scenario* representing an apron activity

Composite_event (Aircraft_Arrival_Preparation,
Physical objects((p1 : Person), (v1 : Vehicle), (z1 : Zone), (z2 : Zone) (z3 : Zone), (z4 : Zone))
Components((c1 : Composite_State GPU_Arrived_In_ERA(v1,z1))
(c2 : Composite_Event GPU_Enters_GPU_Area(v1,z2))
(c3 : Composite_State GPU_Stopped_In_GPU_Are(v1,z2))
(c4 : Composite_State Handler_Gets_Out_GPU(p1, v1,z2, z3))
(c5 : Composite_Event Handler_From_GPU_Deposites_Chocks (p1,v1,z2,z3,z4)))
Constraints((v1->Type = GPU)
(z1->Name = ERA)
(z2->Name = GPU_Area)
(z3->Name = GPU_Door)
(z4->Name = Arrival_Preparation)
(c1 before c2) (c2 before c3) (c3 before c4) (c4 before c5)(c4 during c3) (c5 during c3)))

Scene Understanding Result

SCENARIO AIRCRAFT_ARRIVAL_PREPARATION_SCENARIOS

Vehicle: GPU

Person: Handler

Zones: ERA, GPU_Access, Arrival_Preparation

Dynamic Zone: GPU_Door

Vehicle_Arrived_In_ERA

Gpu_Enters_Gpu_Access_Area

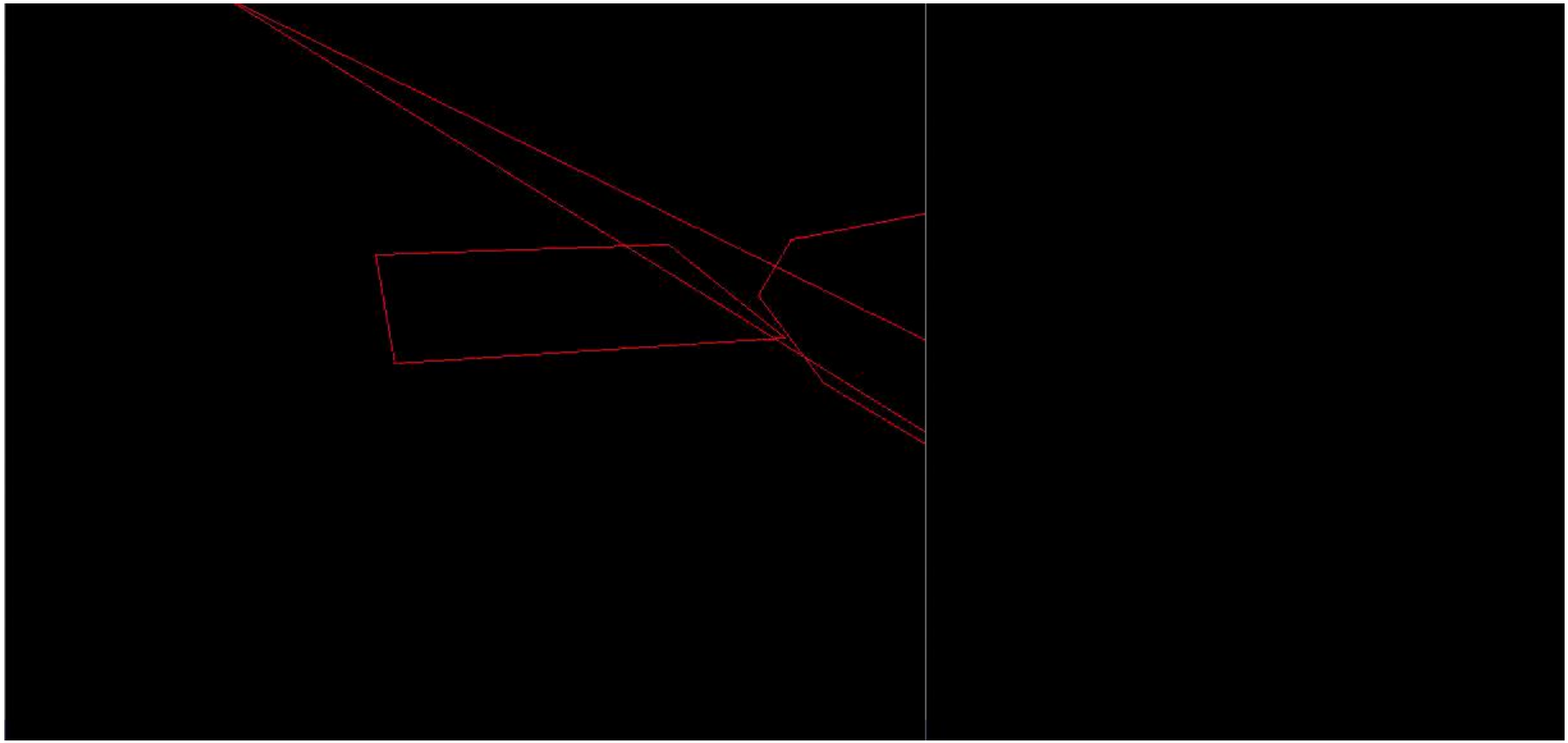
Gpu_Stopped_In_Gpu_Access_Area

Handler_Gets_Out_Gpu

Handler_From_Gpu_Deposites_Chocks_Or_Stud

Preparation of the Aircraft Arrival

Scene Understanding Result



Baggage Loading at the front of the aircraft

Video Event Recognition Algorithm

- **Algorithm in three parts (see VU *et al*, IJCAI'03)**

1. primitive state recognition
2. primitive event recognition given an event template
3. composite state or composite event recognition

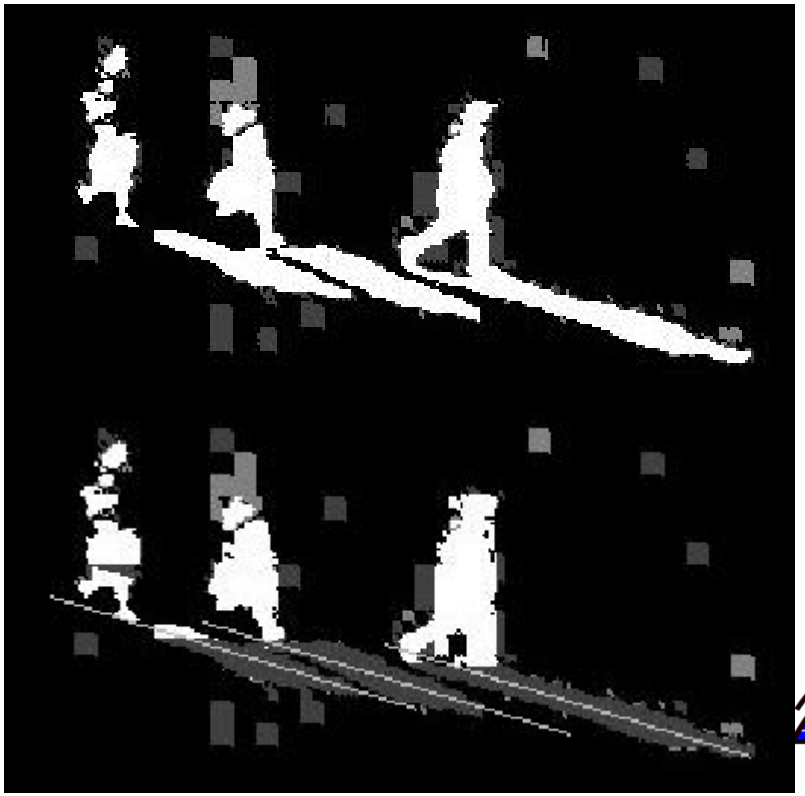
Scene Understanding Results

- **Video Event Modelling:**
  - 28 video events
  - “Aircraft Arrival Preparation”, “Baggage Loading” and “Tanker Arrival” operations
- **Video Event Recognition** tested on 8 video sequences (1899-3774 frames)
    - True Positive : 49
    - False Positive : 0
    - False Negative : 0

Disclaimer: Situations where the tracking module misdetects objects were not addressed

Further Work

- **Motion Detection**: Explicit handling of ghosts and *strong* shadows in visual tracking



Further Work

- **Object Tracking**: added an estimate of the probability of unobstructed observation
- **Object Recognition**: implemented simulated annealing method for top-down model fitting to reduce the likelihood of local maxima detection
- **Object Recognition**: evaluated use of local features (WSMM, harris-laplace) and descriptors (SIFT, NCC) in a bottom-up classifier (currently, a KNN based approach)

Further Work

- **Data Fusion**: Implemented a JPDA filter and extended the validation gate in the standard filter to include location, velocity and category in formation.

- **Data Fusion**: Implemented an epipole based data association algorithm for tracking people off the ground plane.

Further Work

- **Long Term Tracker**: Handling of track ID changes, fragmented tracks, lost tracks and track category confusion.
- **Event and Scenario Recognition**: 58 video events defined and subsequently recognised on representative test data

Future Work

- **Object Tracking**: explicit occlusion analysis to aid reasoning on the congested apron
- **Object Recognition**: improve the robustness of the bottom-up classification stage when objects are interacting
- **Data Fusion**: use particle filter based tracking to improve performance in presence of noise or highly manoeuvering targets
- **Event and Scenario Recognition**: allow uncertainty handling when the tracking result is unreliable

Future Work

- The AVITRACK System: application in other VS domains

Thank you

- **Acknowledgements:**
  EU project **AVITRACK AST3-CT-3002-502818**

- **Website:** www.avitrack.net