Video Understanding

Francois BREMOND

Orion team, INRIA Sophia Antipolis
FRANCE

Key words: Artificial intelligence, knowledge-based systems,
cognitive vision, image understanding,
human behavior representation, scenario recognition

Video Understanding

Definition:

- real time and automated analysis of video sequences
- video understanding= from people detection and tracking to behavior recognition

Recognition of complex behaviors:

of individuals *(fraud, graffiti, vandalism, bank attack)*

of small groups *(fighting)*

of crowds *(overcrowding)*

interactions of people and vehicles *(aircraft refueling)*

Video Understanding Platform

Interpretation of the videos from pixels to alarms

A PRIORI KNOWLEDGE:
- **3d models of the environment**
- **Camera calibration**
- **Scenario Models**

People detection and tracking

People detection and tracking

4 D analysis: multi-cameras tracking

Scenario recognition

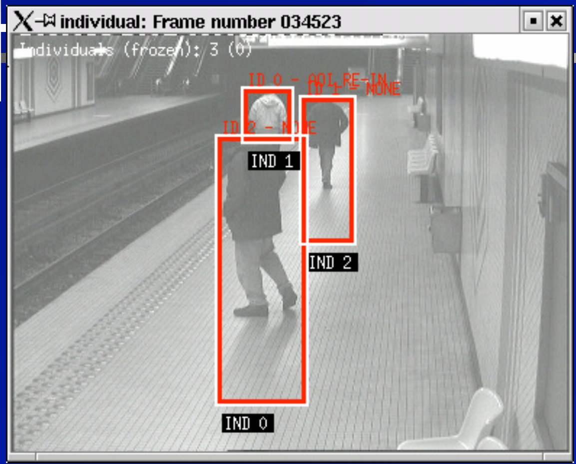
Video understanding

Video Understanding Application

Applications:

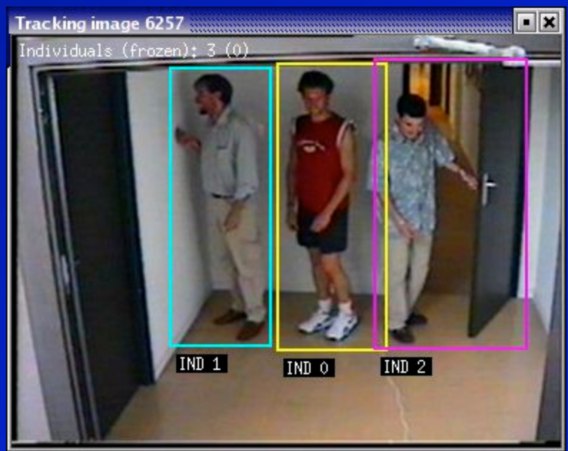
- Strong impact for visual surveillance in transportation (metro station, vehicle traffic, trains, airports)
- Bank agency monitoring
- Control access and Video surveillance in building
- Video understanding for video communication: Mediaspace
- Sports monitoring : swimming pool Surveillance
- New application domains : Aware House, Health (maintaining elderly people at home), Teaching,…

Video Understanding Application

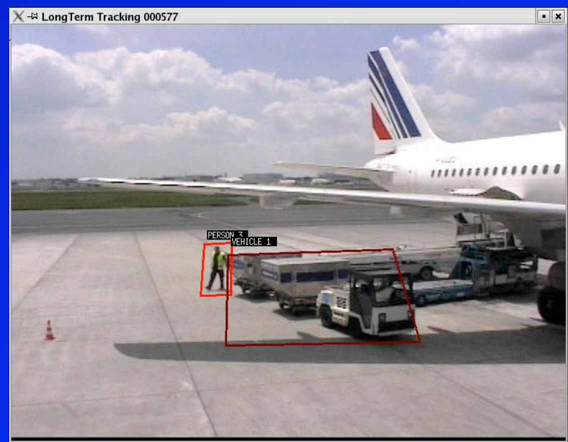


Metro station surveillance

Surveillance inside trains



Building access control



Airport monitoring

Video Understanding: State of the Art

Intelligent Sensors: Acqusition (dedicated hardware), thermic, omni-directional, PTZ, cmos, IP, tri CCD.

Networking : scalable compression, secure transmission and storage.

Computer Vision: Mobile object detection (Wei Yun I2R Singapore), Tracking of people using geometric approaches (T. Ellis Kingston University UK)

Event Recognition: Probabilistic approaches HMM, DBN (A Bobick Georgia Tech USA, H Buxton Univ Sussex UK)

Reusable platform: Realtime video surveillance platform (Multitel, Be), Machine learning

Visualisation: 3D animation, ergonomic, Video abstraction, annotation

Video Understanding: Issues

- Robustness of (vision) algorithms
- Bridging the gaps at different abstraction levels:
  - From sensors to image processing
  - From image processing to 4D analysis
  - From 4D analysis to semantic
- Uncertainty:
  - uncertainty management of noisy data (missing, incomplete, corrupted)
  - formalization of the expertise (fuzzy, incoherent, implicit knowledge)
- Independence of the models/methods versus:
  - sensors and low level preprocessing
  - dedicated applications
  - several spatio-temporal scales

Video Understanding: Issues

- No reliable product on the market
  Solution: adjusting performance to requirement
  First products: traffic control (OCR) and abandoned baggage

- My position: Intelligent Reusable Systems for Cognitive Vision

Intelligent: explicit knowledge, reasoning and learning capabilities
Reusable Systems: different levels of reuse
Cognitive Vision: 4D analysis beyond structural vision (semantics)

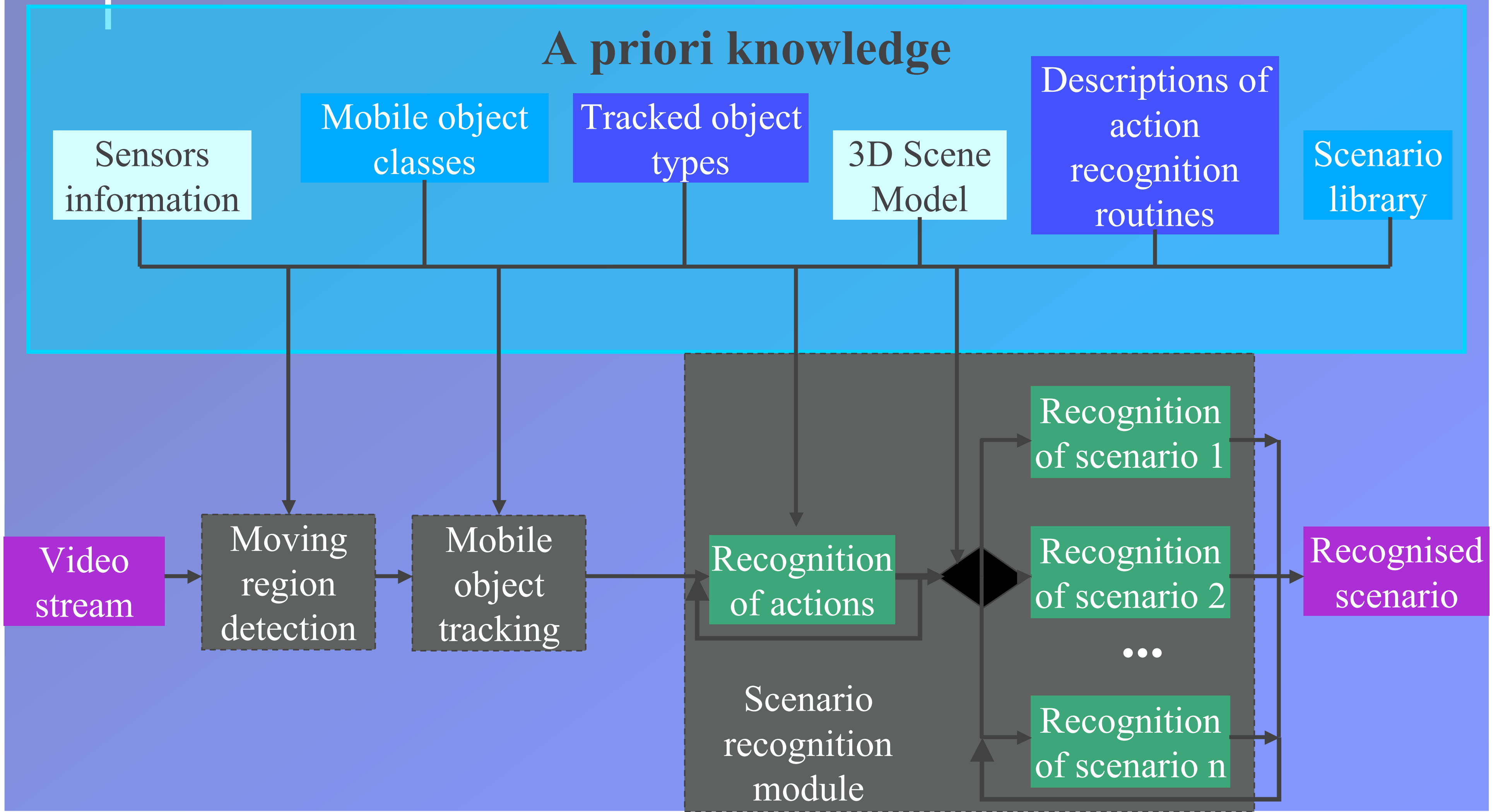
⇨ **Multidisciplinary theme**: artificial intelligence, software engineering, computer vision

Outline

- Introduction on Video Understanding
- Knowledge Representation
- People detection and tracking
- Scenario representation
- Scenario recognition
- Results and conclusion

Knowledge Representation

Knowledge Representation

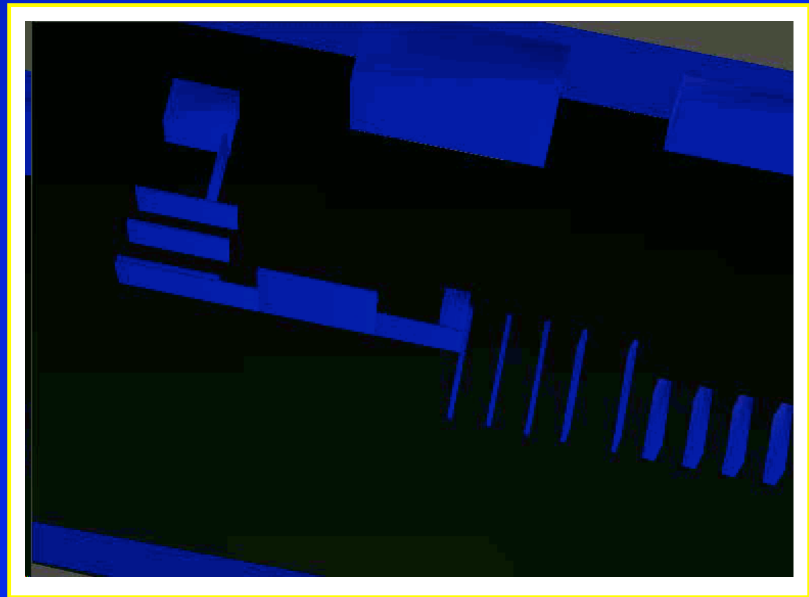


Knowledge Representation: 3D Scene Model

- Definition : a priori knowledge of the observed empty scene
  - Cameras: 3D position of the sensor, calibration matrix, field of view,...
  - 3D Geometry of physical objects (bench, trash, door, walls) and interesting zones (entrance zone) with position, shape and volume
  - Semantic information : type (object, zone), characteristics (yellow, fragile) and its function (seat)
- Role:
  - to keep the interpretation **independent** from the sensors and the sites : many sensors, one 3D referential
  - to provide **additional knowledge** for behavior recognition

Knowledge Representation: 3D Scene Model

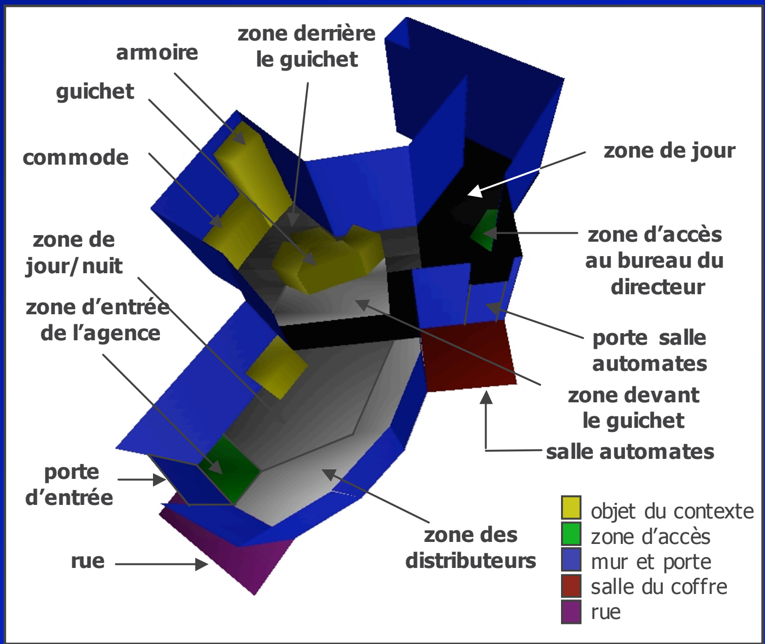
- Barcelona Metro Station Sagrada Famiglia mezzanine (cameras C10, C11 and C12)



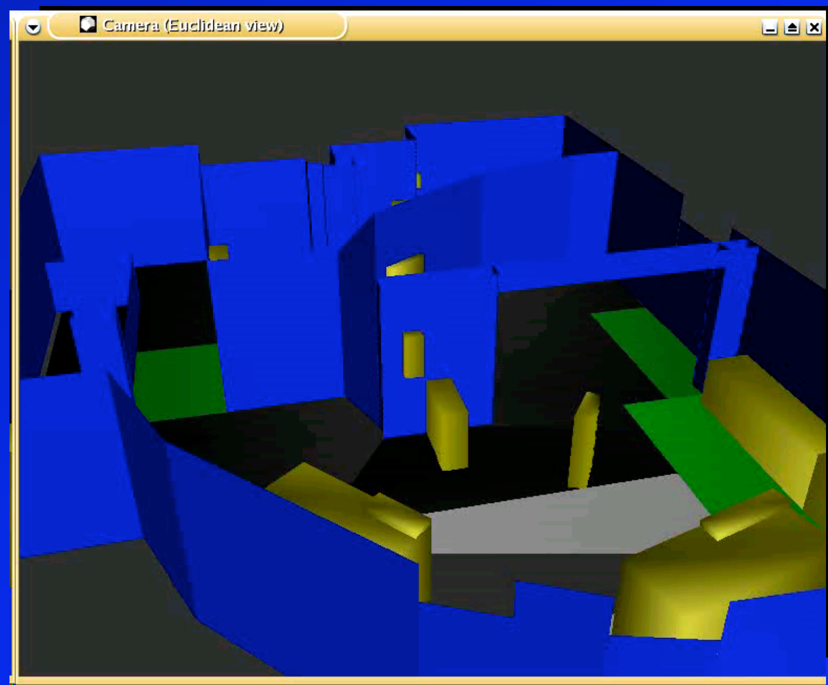
Knowledge Representation : 3D Scene Model

3d Model of 2 bank agencies

Les Hauts de Lagny

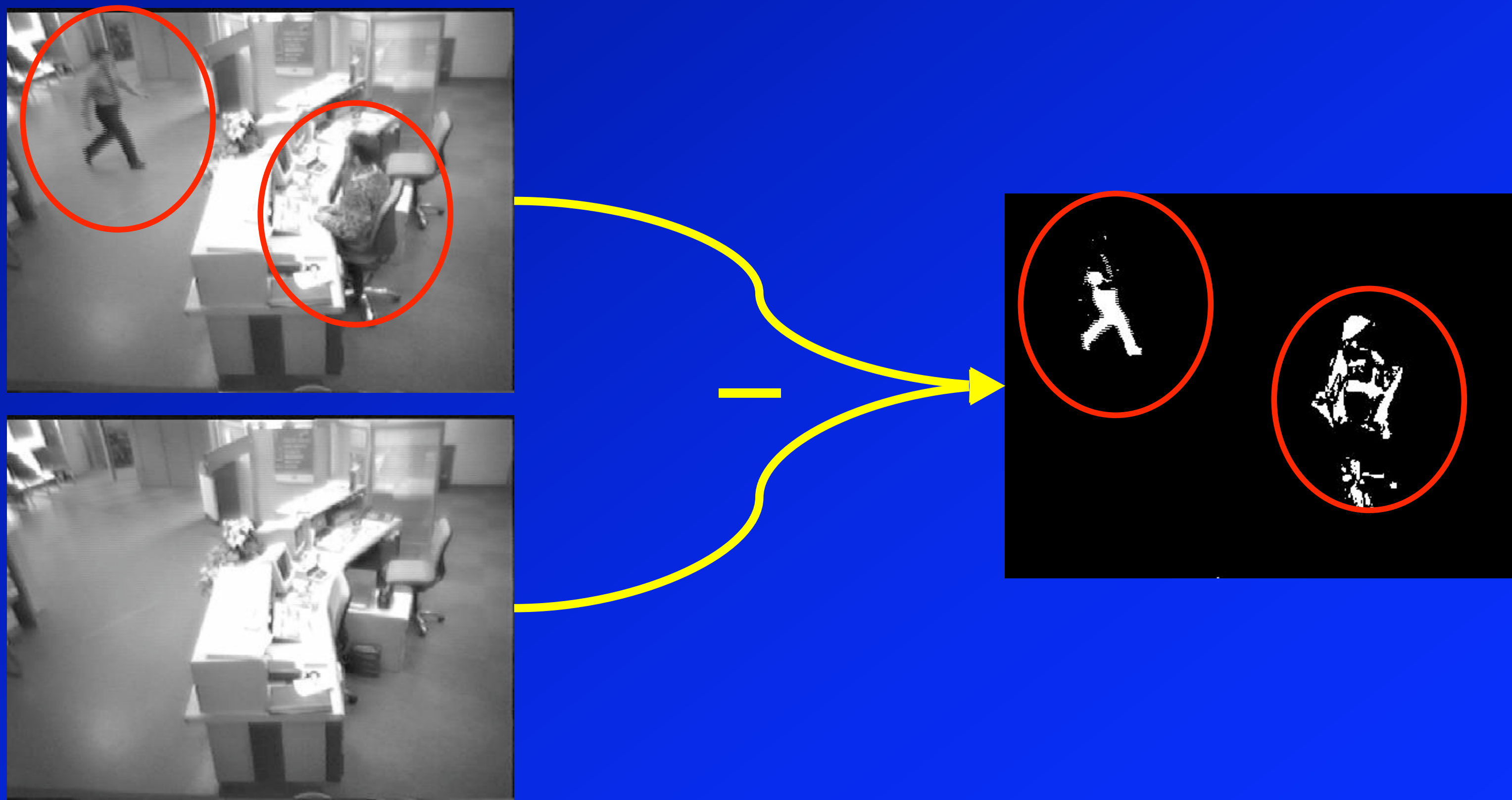


Villeparisis



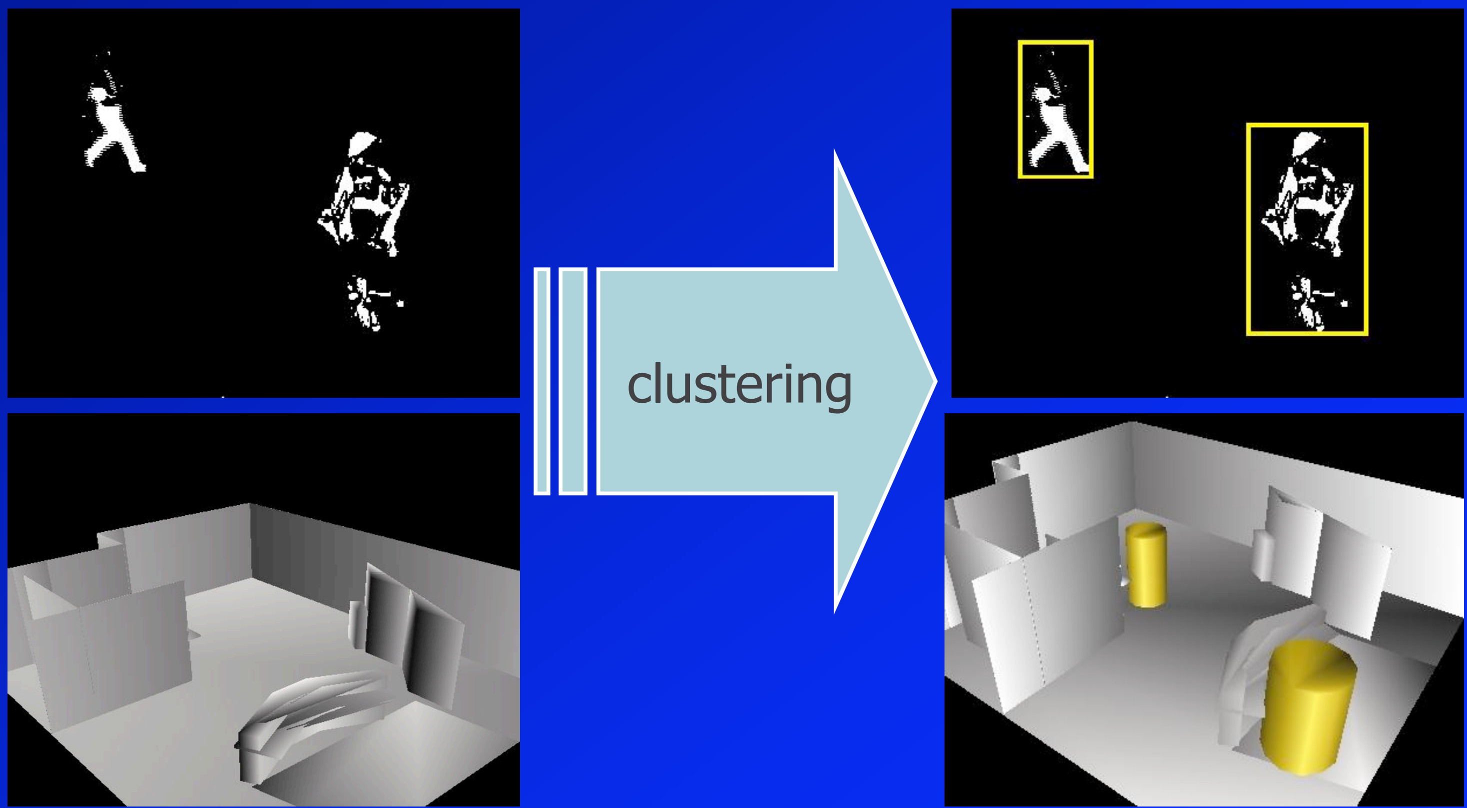
People detection and tracking

- Difference between the current image and a reference image (computed) of the empty scene



People detection and tracking

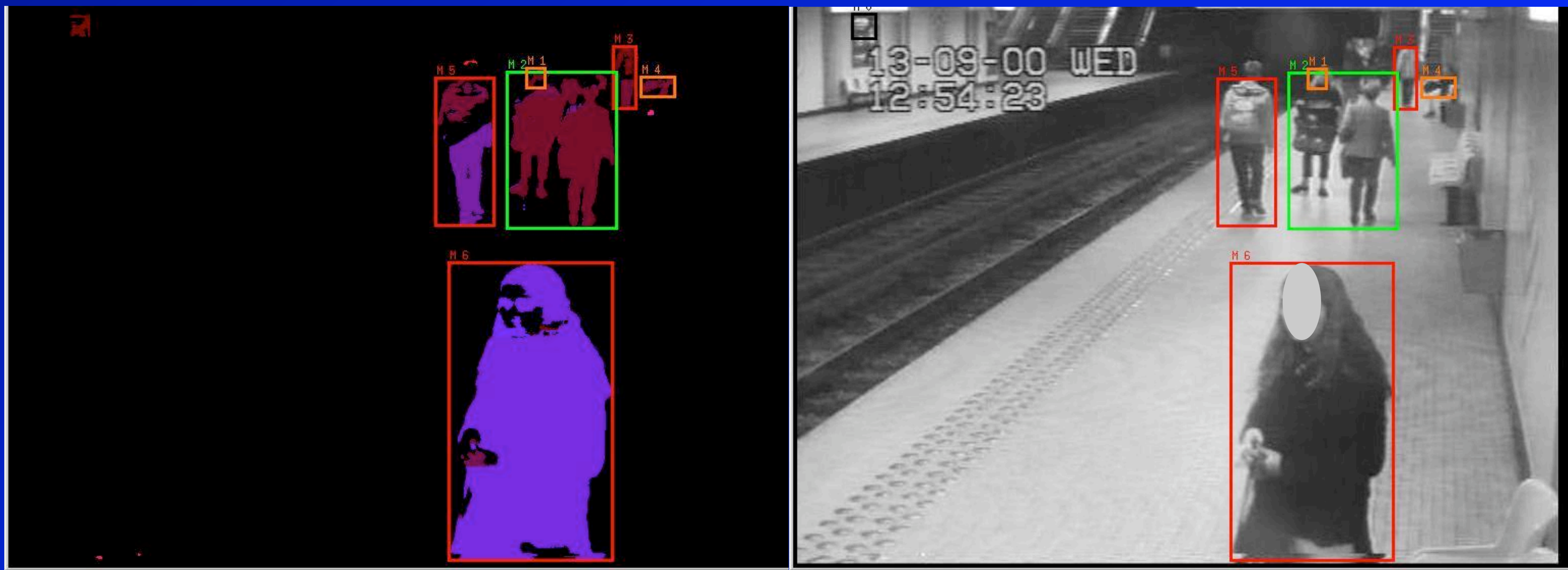
Approach: Group the moving regions together to obtain a bigger moving region matching a mobile object model



People detection and tracking

Classification into more than 8 classes (e.g. Person, Groupe, Train) based on 2D and 3D descriptors (position, width and height)

Example of 4 classes: Person, Group, Noise, Unknown



People detection and tracking

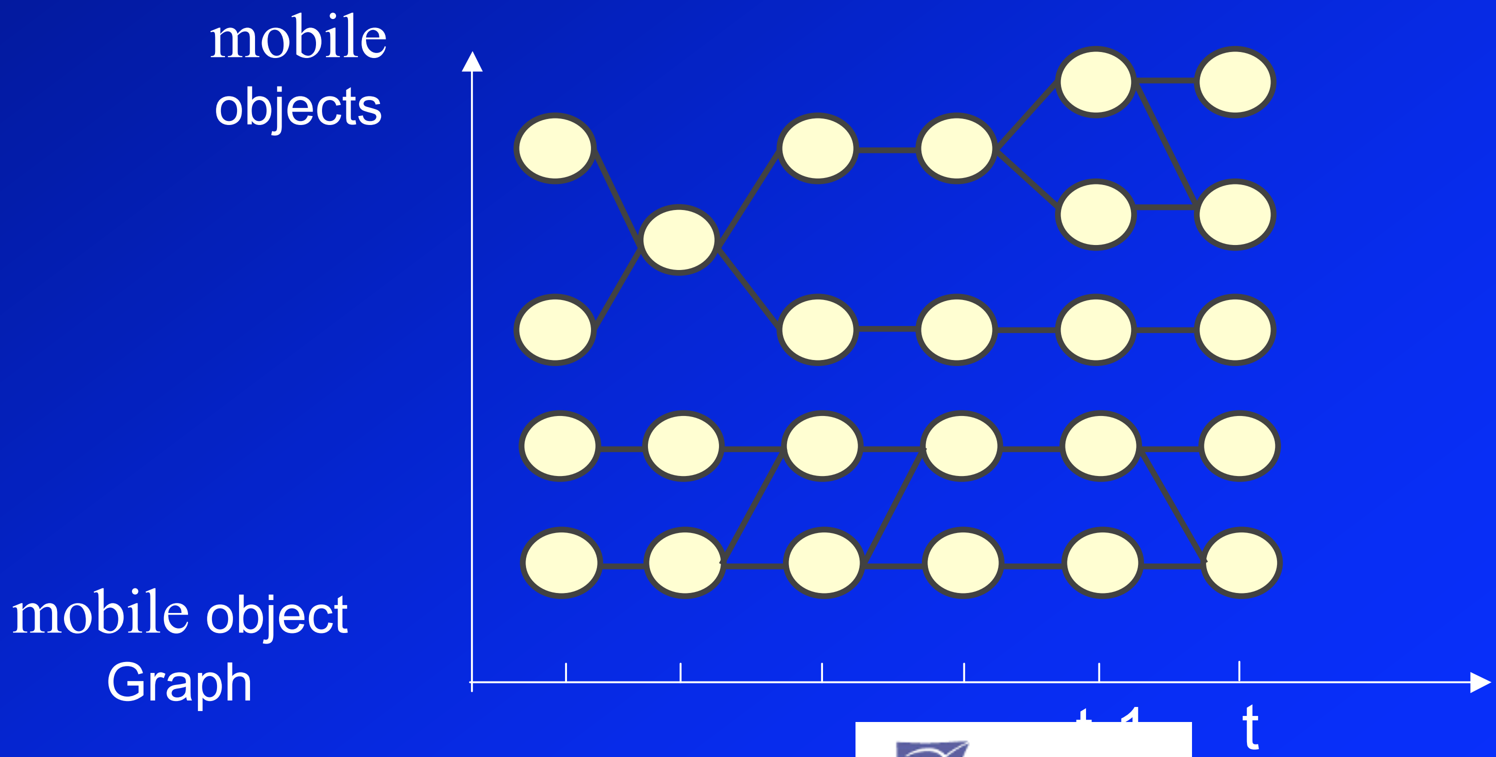
Approach: A Computation of correspondences between the moving regions newly detected at t and the moving regions already detected at t-1 using three criteria (2D and 3D distance and similitude).

At time t-1:

At time t:

People detection and tracking

- **Frame to frame tracking:** For each image all newly detected moving regions are associated to the old ones through a graph



People detection and tracking

objet: mobile personne

INDIVIDU suivi

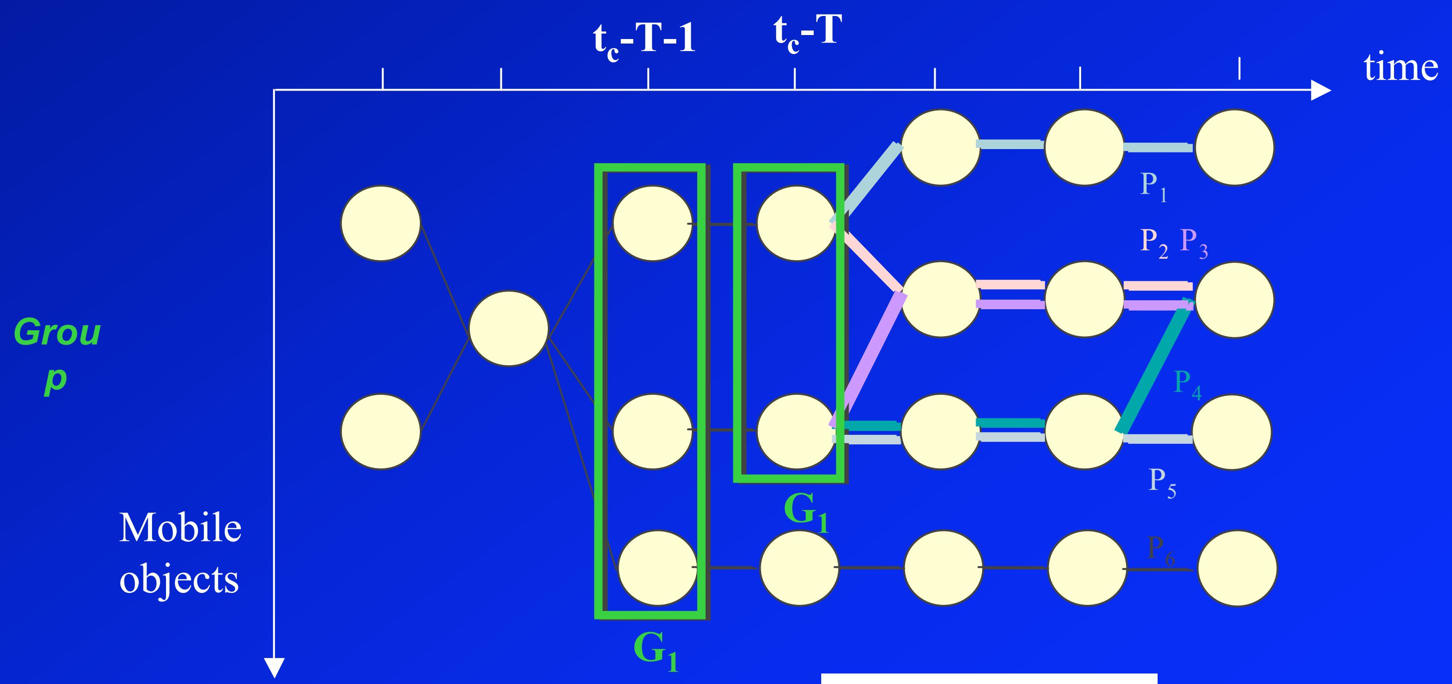
Limitations : - Mixed of individuals in difficult situations (e.g. static and dynamic occlusion, long crossing)

People detection and tracking

Goal : To track globaly people over a long time period

Method: Analysing of the mobile object graph

Group model, Model of trajectories of people inside a group, use of time delay



People detection and tracking

mobile object: Person

mobile object: Group

mobile object: Unkown

mobile object: Occluded-person

mobile object: Person?

mobile object: Noise

Tracked ***GROUP***

Limitations : - Imperfect estimation of the group size and location when there are shadows or reflections strongly contrasted.

- Imperfect estimation of the number of persons in the group when the persons are occluded, overlapping each others or in case of miss detection.

Scenario Representation

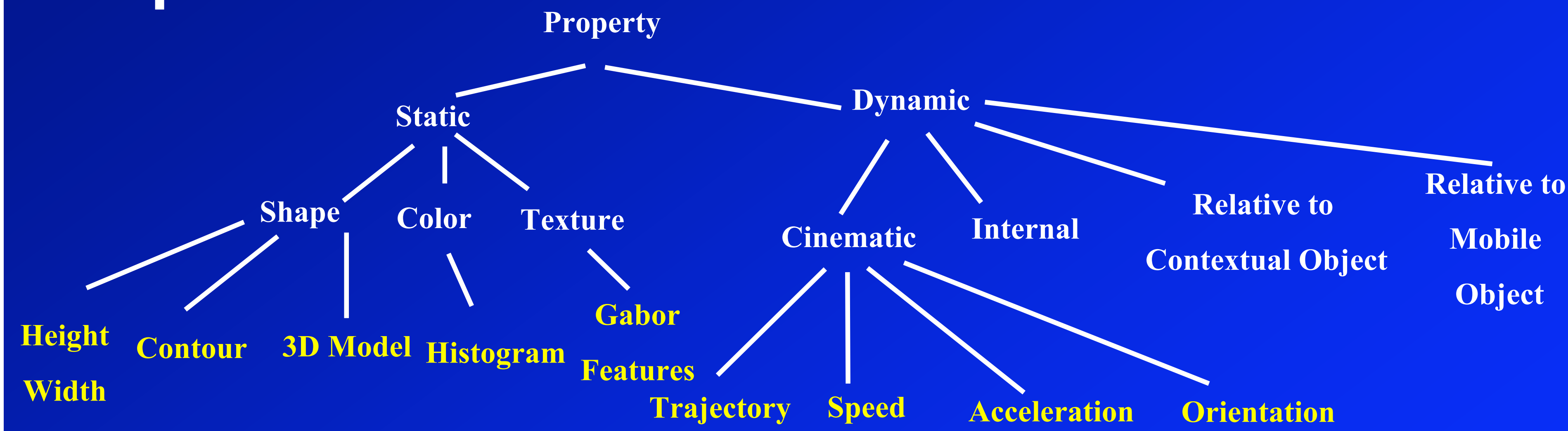
Scenario Representation

We define several entities:

- **Context object**: predefined static object of the scene environment (entrance zone, bench, walls, equipment,...).
- **Moving region**: any intensity change between a reference and the current images.
- **Mobile object**: any moving region which has been tracked and classified (e.g. person, group of persons, part of human body, door).
- **Mobile object Property** on one or several mobile objects

Scenario Representation

Properties :



➡ **Properties are close to vision features and are computed by vision algorithms**

Scenario Representation

- **Issues**: **large variety of actions and scenarios**
  - more or less **abstract** (running/fighting).
  - general (standing)/sensor and application (sit down) dependent.
  - **spatial granularity**: the view observed by one camera/the whole site.
  - **temporal granularity**: instantaneous/long term.
  - **3 levels of complexity** depending on the complexity of temporal relations and on the number of actors:
    - non-temporal constraint relative to one actor (being seated).
    - temporal sequence of sub-scenarios relative to one actor (open the door, go toward the chair then sit down).
    - complex temporal constraints relative to several actors (A meets B at the coffee machine then C gets up and leaves).

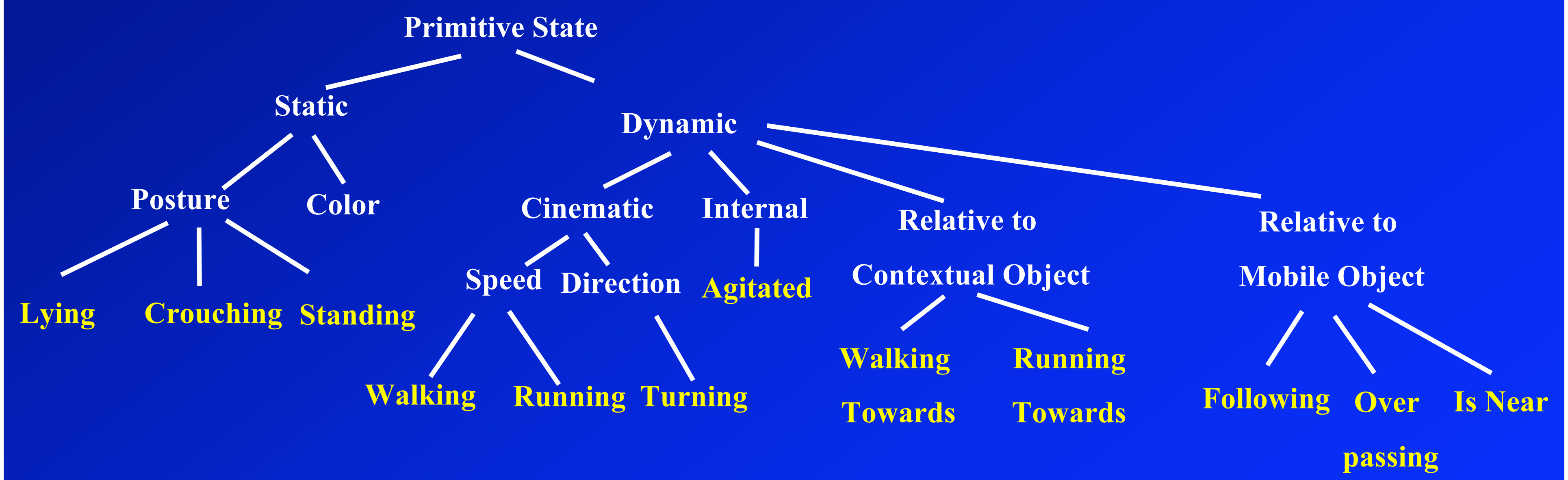
Scenario Representation

- *Video events (real world notion):*
  - *Primitive State:* a spatio-temporal property linked to vision routines involving one or several actors, valid at a given time point or stable on a *time interval* (a coherent unit of motion of a mobile object). *Ex : « close», « walking», « seated»*
  - *Composite State: a combination of primitive states*
  - *Primitive Event:* a significant change of states

    Ex : « enters», « stands up», « leaves »
  - *Composite Event:* a combination of states and events. Corresponds to a long term (symbolic, application dependent) activity. Ex : « fighting», « vandalism»

Scenario Representation

Primitive State



Scenario Representation

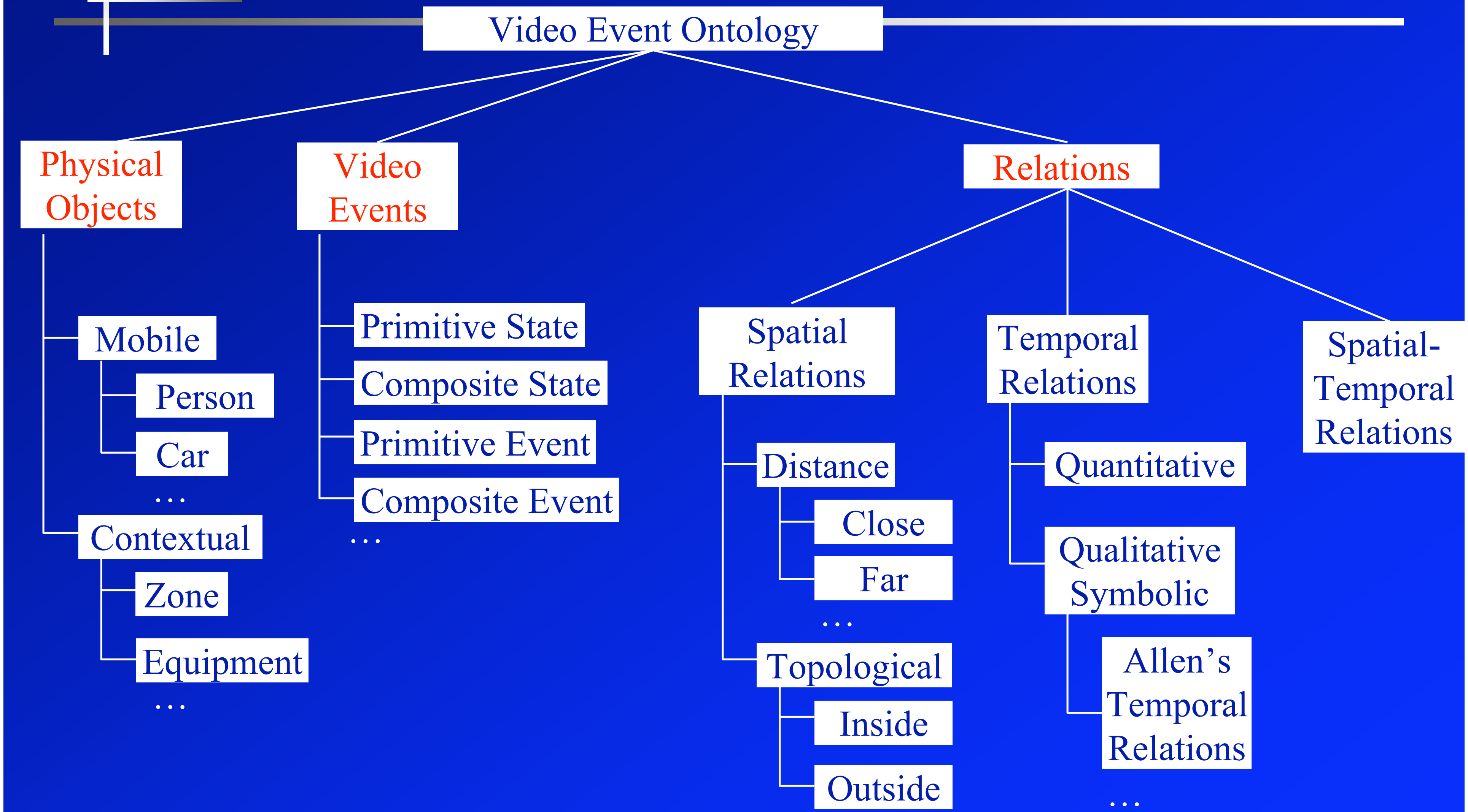
Scenario (algorithmic notion): any type of video events

- Two types of scenarios: **elementary** and **composed**.
- Elementary scenario: (primitive state) spatio-temporal property valid at a given instant or stable on a time interval.
  Example: “Inside_zone”.
- Composed scenario is a complex video event valid at a given instant or stable on a time interval.
  Example: “Bank_attack”.

Scenario Representation

- Video Event Ontology: a set of concepts and relation is used as a reference between all the actors of the domain to describe knowledge
- Enable experts to describe video events, (e.g. primitive state, composite event): ontology of the application domain.
- Share knowledge between developers: ontology of visual concepts (e.g. a stopped mobile object)
- Ease communication between developers and end users and enable performance evaluation: ontology of the video understanding process (what should be detected: mobile object (a parked car), object of interest (a door), visible object (occluded person))
- Architecture interoperability: separation between system interface and knowledge description

Scenario Representation



Scenario Representation

Meta-Model of Scenarios

Physical Objects

- a set of **variables** corresponding to physical objects relative to the scenario

Components

- a set of **variables** corresponding to the components composing the scenario
- only for **composed** scenario models

Forbidden Components

- a set of **variables** corresponding to forbidden components
- only for **composed** scenario models

Constraints

- **symbolic**, **logical** and **spatial** constraints are used
- **temporal** constraints are used to define **composed** scenario models

Actions

- a set of **tasks** to be performed when the scenario is recognized

Scenario Recognition

- A scenario is mainly constituted of three parts :
  - Physical objects: all real world objects present in the scene observed by the cameras
    - Mobile objects, contextual objects, zones of interest
  - Components: list of states and events involved in the scenario
  - Constraints: symbolic, logical, spatio-temporal relations between components or physical objects

Scenario Representation

Example: “inside_zone” & “changes_zone” scenario models

primitive-state(*inside_zone*,
physical-objects((p : **Person**), (z : **Zone**))
constraints((p *in* z)))

primitive-event(*changes_zone*,
physical-objects((p : **Person**), (z1 : **Zone**), (z2 : **Zone**))
components(
(e1 : primitive-state *inside_zone*(p, z1))
(e2 : primitive-state *inside_zone*(p, z2)))
constraints((e1 *before* e2)))

Scenario Representation

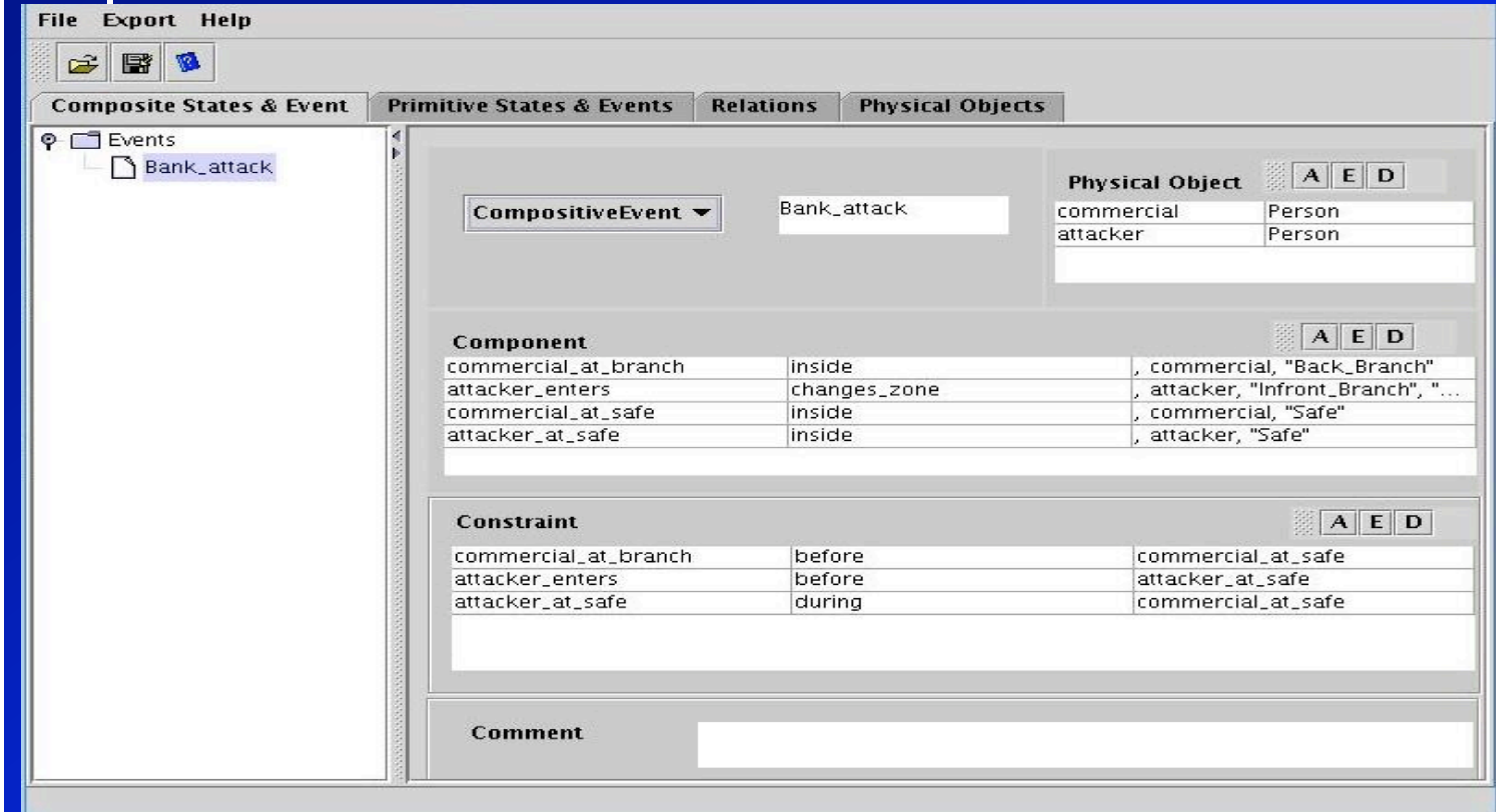
A “Bank attack” scenario instance

Scenario Representation

Example: a “Bank_Attack” scenario model

```
composite-event(Bank_attack,
   physical-objects((employee : Person), (robber : Person))
   components(
         (e1 : primitive-state inside_zone(employee, "Back"))
         (e2 : primitive-event changes_zone(robber, "Entrance", "Infront"))
         (e3 : primitive-state inside_zone(employee, "Safe"))
         (e4 : primitive-state inside_zone(robber, "Safe"))   )
    constraints((e2   during    e1)
                (e2   before   e3)
                (e1   before   e3)
                (e2   before   e4)
                (e4 during   e3)  )
    alert(“Bank attack!!!”) )
```

Scenario Representation



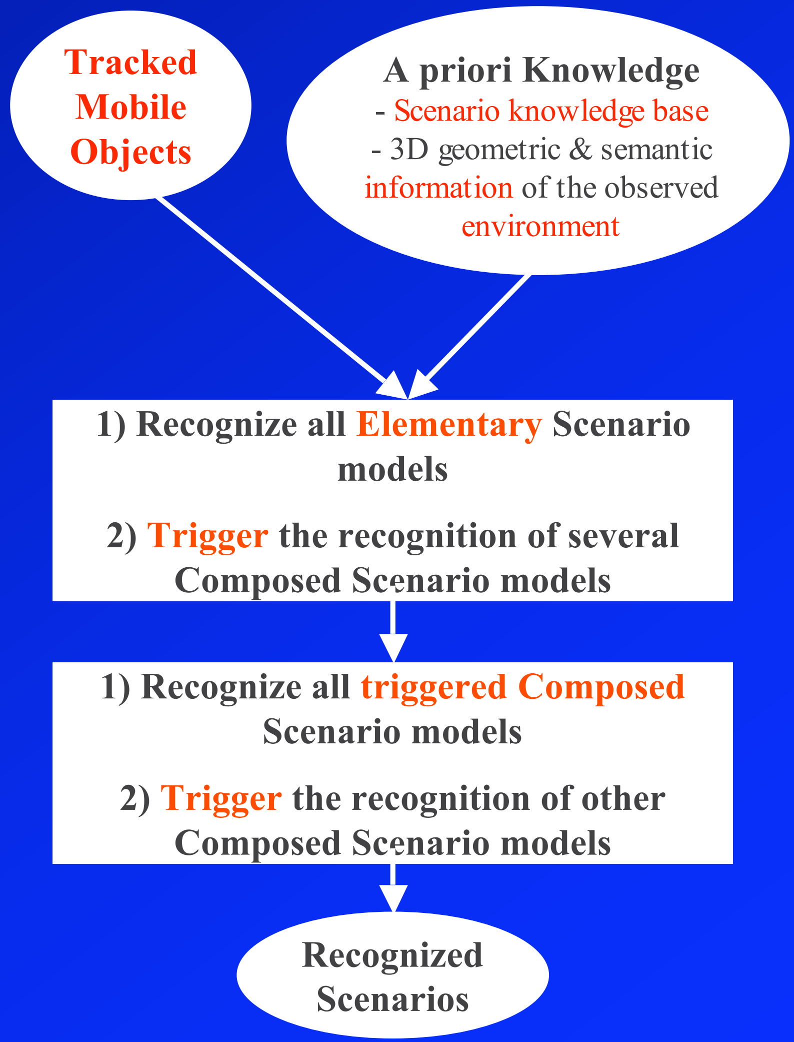
Scenario Recognition

Scenario Recognition

Overview of Recognition Process

Scenario recognition algorithm at each instant:

1. Recognize the elementary scenario models ➔ store scenario instances and trigger the recognition of composed scenario models.
2. Recognize the composed scenario models ➔ store scenario instances and trigger the recognition of other composed scenario models.



Scenario Recognition: Elementary Scenario

- The recognition of a compiled elementary scenario model m_e consists of a loop:

  1. Choosing a physical object for each physical-object variable
  2. Verifying all constraints linked to this variable
  3. m_e is recognized if all the physical-object variables are assigned a value and all the linked constraints are satisfied.

Scenario Recognition: Composed Scenario

- **Problem**: Temporal Constraint Resolution

  given a scenario model m_c = (m_1 *before* m_2 *before* m_3); if a scenario instance ρ_3 of m_3 has been recognized → it makes sense to try to recognize the main scenario model m_c. However, the classical algorithms will try all combinations of scenario instances of m_1 and of m_2 with ρ_3 → a combinatorial explosion.

- **Solution**: decompose the composed scenario models into simpler scenario models in an initial stage (compilation of composed scenario models) such as each composed scenario model is composed of two components.

Scenario Recognition: Composed Scenario

Example: original “Bank_attack” scenario model

composite-event(***Bank_attack***,
  physical-objects((employee : **Person**), (robber : **Person**))
  components(
    (1)   (e1 : primitive-state ***inside_zone***(employee, "Back"))
    (2)   (e2 : primitive-event ***changes_zone***(robber, "Entrance", "Infront"))
    (3)   (e3 : primitive-state ***inside_zone***(employee, "Safe"))
    (4)   (e4 : primitive-state ***inside_zone***(robber, "Safe")))
  constraints((e2 ***during*** e1)
      (e2 ***before*** e3)
      (e1 ***before*** e3)
      (e2 ***before*** e4)
      (e4 ***during*** e3))
  alert(“Bank attack!!!”))

Scenario Recognition: Composed Scenario

Compilation: Original scenario model is decomposed into 3 new scenarios

```
composite-event(Bank_attack_1,
    physical-objects((employee : Person), (robber : Person))
    components(
    (1)     (e1 : primitive-state inside_zone(employee, "Back"))
    (2)     (e2 : primitive-event changes_zone(robber, "Entrance", "Infront"))
    constraints((e1 during e2)   ))

composite-event(Bank_attack_2,
    physical-objects((employee : Person), (robber : Person))
    components(
     (3)    (e3 : primitive-state inside_zone(employee, "Safe"))
     (4)    (e4 : primitive-state inside_zone(robber, "Safe"))     )
    constraints((e3 during e4)    ))

composite-event(Bank_attack_3,
    physical-objects((employee : Person), (robber : Person))
    components(
            (att_1 : composite-event Bank_attack_1(employee, robber))
            (att_2 : composite-event Bank_attack_2(employee, robber))    )
    constraints(((termination of att_1) before (start of att_2))   )
    alert("Bank attack!!!")      )
```

Scenario Recognition: Composed Scenario

- A compiled scenario model m_c is composed of two components: start and termination.
- To start the recognition of m_c, its termination needs to be already instantiated.
- The recognition of a compiled scenario model m_c consists of a loop:
  1. Choosing a scenario instance for the *start* of m_c,
  2. Verifying the temporal constraints of m_c,
  3. Instantiating the physical-objects of m_c with physical-objects of the *start* and of the *termination* of m_c,
  4. Verifying the non-temporal constraints of m_c.
  5. Verifying forbidden constraints.

Scenario recognition: capacity of prediction

- Issue: in the bank monitoring application, an alert "Bank attack!!!" is triggered when a scenario "Bank_attack" is recognized. However, it can be too late for security agents to cope with the situation.

- Requirement: is the temporal scenario recognition method able to predict scenarios that may occur in the future?

- Answer: YES, the recognition algorithm can predict scenarios that may occur by adding automatically alerts (during the compilation) to some generated scenario models. This task can be specified in scenario models (precursor events).

Scenario recognition : uncertainty

- Temporal tolerance
  - Issue: several scenario models are defined with too strong temporal constraints ⇒ they cannot be recognized with real videos.
  - Solution: we defined a temporal tolerance Δ_t as an integer, then all temporal comparisons are estimated using an approximation of Δ_t.

- Incorrect mobile object tracking
  - Issue: the vision algorithms may loose the track of several detected mobile objects ⇒ the system cannot recognize correctly scenario occurrences in several videos.
  - Solution: experts describe different scenario models representing various situations corresponding to several combinations of physical objects.

Scenario recognition : learning

Several types of techniques to learn additive knowledge:

- At a scenario level:
  - the parameters to tune specific recognition routines
  - the best recognition routines (e.g. classifier structure...)
  - the most relevant features (e.g. contours) to support efficient recognition dedicated to a given property, state, event or scenario
- At the system level:
  - to get an automatic set-up (calibration, reference image, 3D scene model)
  - to choose dynamically the most appropriate routines
  - to learn regular temporal patterns corresponding to scenarios

Issues:

- **building learning/test sets and ground truth**
- **need of ontology for the learning/test sets**

Scenario Recognition: Results

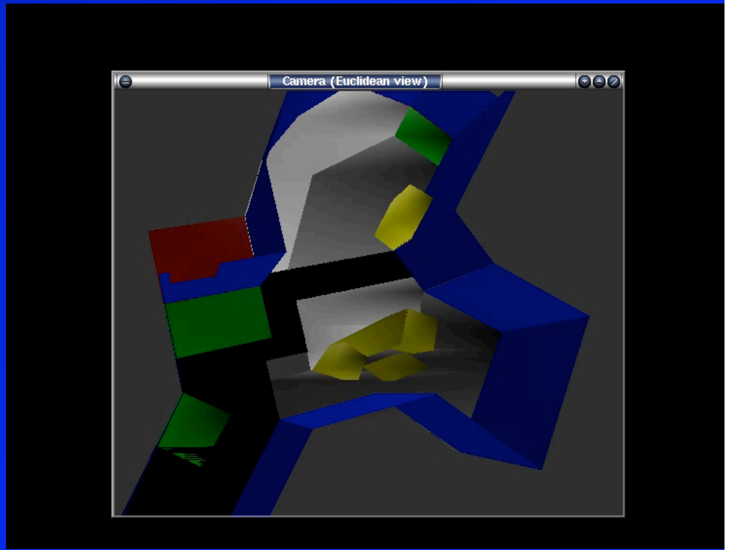
Scenario recognition: Results

The experts of four projects in video interpretation (CASSIOPEE -bank monitoring-, ADVISOR -metro surveillance-, AVITRACK -apron monitoring- and SAMSIT -inside train surveillance-) have used the Orion system in different sites to realize three types of tests.

- on recorded videos: to verify whether the recognition algorithm can recognize sufficiently scenario occurrences.
- on live videos: to verify whether the recognition algorithm can work in a longtime interval.
- on recorded/simulated videos: to estimate the processing time of the recognition algorithm.

Scenario recognition: Results

Results in bank agency



Scenario recognition: Results

Results in bank agency

People Counting scenario

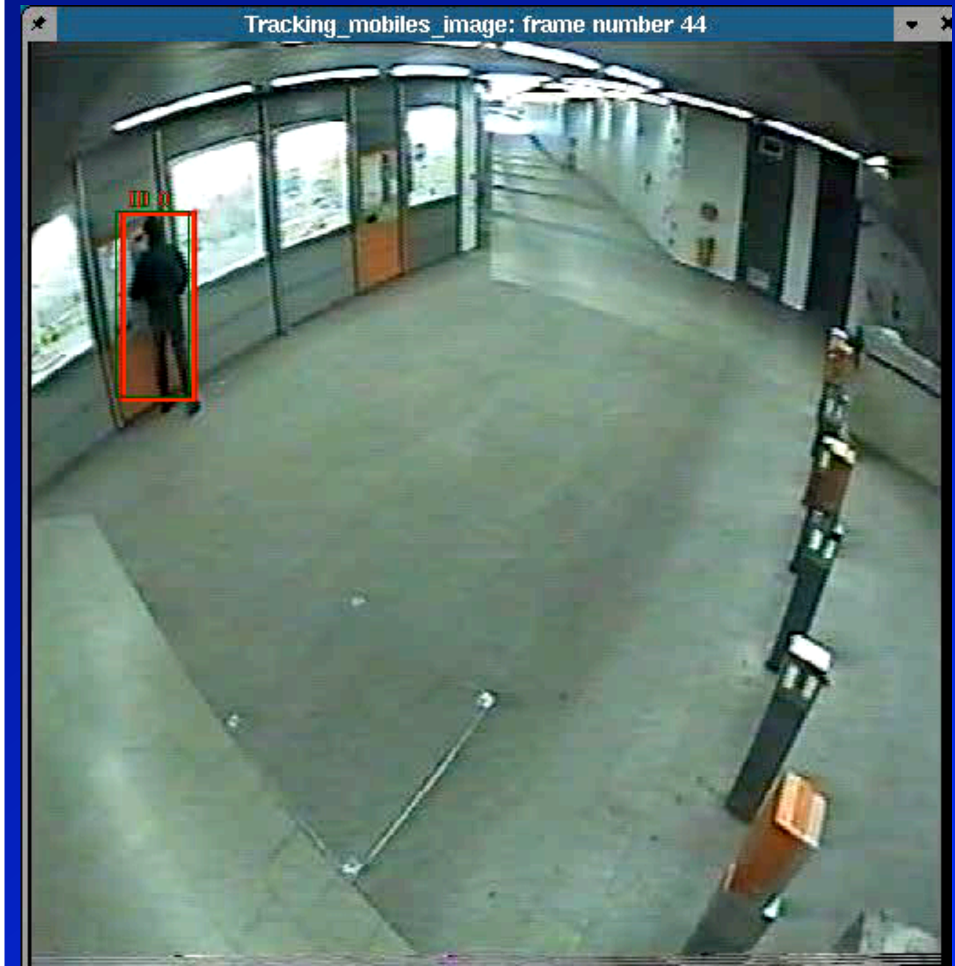
Scenario recognition: Results

- Vandalism scenario example (temporal constraints) :

Scenario(vandalism_against_ticket_machine,
Physical_objects((p : **Person**), (eq : **Equipment**, ***Name***="Ticket_Machine"))
Components ((**event** s1: *p* **moves_close_to** *eq*)
(**state** s2: *p* **stays_at** *eq*)
(**event** s3: *p* **moves_away_from** *eq*)
(**event** s4: *p* **moves_close_to** *eq*)
(**state** s5: *p* **stays_at** *eq*))
Constraints ((s1 **!=** s4) (s2 **!=** s5)
(s1 **before** s2) (s2 **before** s3)
(s3 **before** s4) (s4 **before** s5))))

Scenario Recognition: Results

Vandalism in metro (Nuremberg)



Scenario recognition: Results

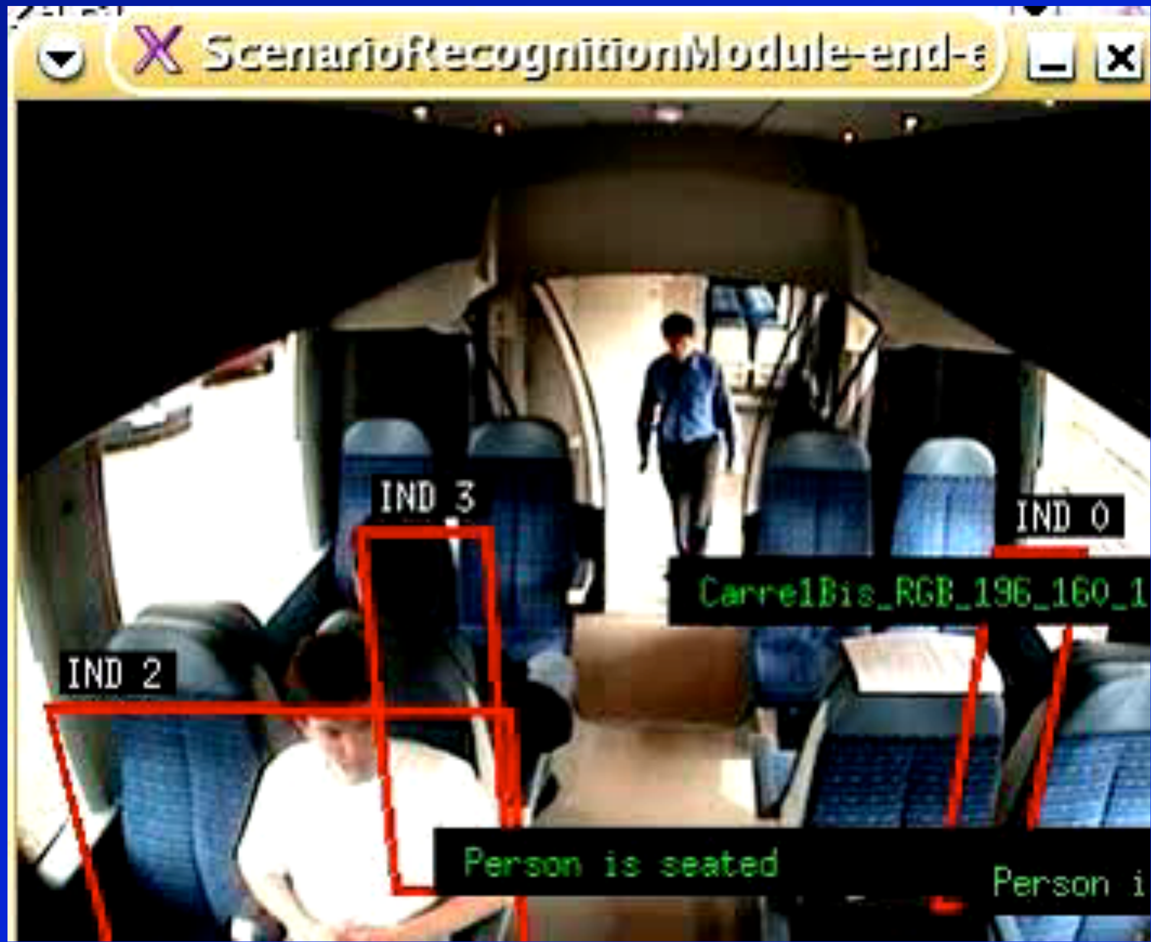
Example: a "Vandalism against a ticket machine" scenario

Scenario recognition: Results
Example: a "Aircraft Tanker" event

Scenario recognition: Results
Example: "Aircraft GPU / Loader" event

Scenario recognition: Results
Example: "Scratch & theft in a train" scenarios

Scenario recognition: Results
Example: a “Disturbing people in a train” scenario



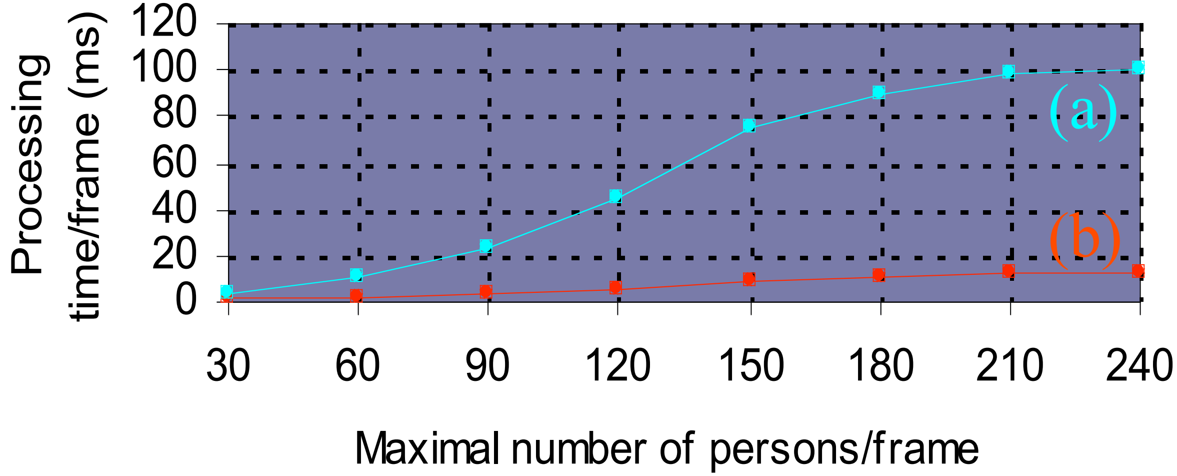
Scenario recognition: Results
Experiment 2: live-videos

- 6 sites: 2 bank agencies, two offices, a parking and a metro station.
- 40 original scenario models (before decomposition): “inside_zone”, “Bank_attack”, “Vandalism”,...
- Results:
  - in a bank (5 days),
  - in an office (4h),
  - one week in a metro station of Barcelona,
  - in a parking (1 day)
  - the scenarios were most of the time (95%) correctly recognized (as in the first experiment) ➔ the recognition algorithm can work reliably and robustly in real-time and in a continuous mode.

Scenario recognition: Results

Experiment 3: checking the processing time

60 scenario models defined with 2 to 10 physical object variables and 2 to 10 components. The algorithms are tested on simulated videos containing up to 240 persons in the scene.



The (b) **average** and (a) **maximal** processing time/frame of the algorithm.

The composed scenario recognition algorithm is able to process up to 240 persons in the scene.

Scenario Recognition: Conclusion

Conclusion

a global framework for video surveillance:

- Hypotheses:
  - fixed cameras
  - 3D model of the empty scene
  - predefined behavior models
- Results:
  - Behavior understanding for Individuals, Groups of people or Crowd
  - an operational language for video understanding (more than 50 states and events)
  - a real-time platform (5 to 25 frames/s)

Conclusion: issues

Knowledge Acquisition

- Design of ontology driven knowledge acquisition:
  - *video event ontology*
- Design of learning techniques to complement a priori knowledge:
  - *visual concept learning*
  - *scenario model learning*

Video event detection

- Finer human shape description:*3D posture models*
- Video analysis robustness*: Uncertainty management*

Reusability is still an issue for vision programs

- The vision module cannot always cope with the real world complexity
- Use of program supervision techniques: *dynamic configuration of programs and parameters*
- Scaling issue: managing large network of heterogeneous sensors (cameras, microphones, optical cells, radars….)

Video Understanding

Francois BREMOND

Orion team, INRIA Sophia Antipolis, FRANCE
http://www-sop.inria.fr/orion/

Key words: Artificial intelligence, knowledge-based systems, cognitive vision, image understanding, human behavior representation, scenario recognition